

BAB 6

PENUTUP

6.1 Kesimpulan

Berdasarkan dari pembahasan-pembahasan pada bab-bab sebelumnya, maka dapat ditarik kesimpulan dari Tugas Akhir ini :

1. Aplikasi MEPAMS berhasil dibangun untuk membantu pelanggan Pamella Swalayan dalam melakukan transaksi pemesanan yang berjalan pada smartphone Android menggunakan *ionic framework* dan *angularJS*.
2. Aplikasi MEPAMS mampu menerapkan *Market Basket Analysis* sebagai metode yang memberikan rekomendasi produk kepada pelanggan pada aplikasi *mobile*.

6.2 Saran

Setelah melalui proses analisis, perancangan, implementasi, hingga pengujian aplikasi maka ada beberapa saran untuk pengembangan aplikasi MEPAMS, yaitu:

1. Pengembangan aplikasi dapat dikembangkan dengan fitur *location based services* dan *notifikasi*.
2. Fitur rekomendasi dapat dikembangkan kembali dengan metode yang berbeda selain *Market Basket Analysis*.

DAFTAR PUSTAKA

Aguinis, H., Forcum, L. & Joo, H., 2013. Using Market Basket Analysis in Management Research. *Journal of Management*, 39(7), pp. 1779-1824.

Boedijono, H., Andjarwirawan, J. & Setiawan, A., 2015. Pembuatan Aplikasi News Dwi Pekan Universitas Kristen Petra Berbasis Android. *Jurnal Infra*, 3(2), pp. 196 - 202.

Budiardjo, E. K. & Irwiensyah, F., 2008. *Analisis Fitur CRM Untuk Meningkatkan Kepuasan Pasien Berbasis Pada Framework Of Dynamic CRM*. Yogyakarta, Seminar Nasional Informatika 2008. Yogyakarta, 2008. Jurusan Teknik Informatika UPN "Veteran" Yogyakarta.

Eva, A., 2007. *PERSEPSI PENGGUNAAN APLIKASI INTERNET UNTUK PEMASARAN PRODUK USAHA KECIL MENENGAH*. Yogyakarta, Seminar Nasional Aplikasi Teknologi Informatika. Yogyakarta, 2007. Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Evy, N., 2012. Analisis Dan Perancangan Web Server Pada Handphone. *Jurnal Sistem Informasi*, 5(2), pp. 1-17.

Fatmanto, A.H., 2013. *Pembangunan Aplikasi Forum Jual Beli Berbasis Mobile*. Skripsi. Yogyakarta: Universitas Atma Jaya Yogyakarta Universitas Atma Jaya Yogyakarta.

Fuad, E., Ferdiana, R. & Selo, 2014. *PERANCANGAN FITUR E-COMMERCE BERDASARKAN KONSEP CUSTOMER RELATIONSHIP MANAGEMENT UNTUK MENINGKATKAN KUALITAS PELAYANAN*. Kudus, Prodising SNATIF. Kudus, 2014. Fakultas Teknik, Universitas Muria Kudus

Irmawati, D., 2011. Pemanfaatan E-Commerce Dalam Dunia Bisnis. *Jurnal Ilmiah Orasi Bisnis*, IV(ISSN:2085-1375), pp. 95-112.

Koepradono, Suraya & Rachmawati, Y., 2013. Sistem Informasi Pengolahan Data Pertumbuhan Ekonomi dan

Ketimpangan Di Kabupaten Klaten(Tahun 2013-2012) Menggunakan Framework Codeigniter. *Jurnal Script*, 1(1), pp. 46-54.

Lengkong, H. N., Sinsuw, A. A. & Lumenta, A. S., 2015. Perancangan Penunjuk Rute Pada Kendaraan Pribadi Menggunakan Aplikasi GIS Berbasis Android Yang Terintegrasi Pada Google Maps. *E-journal Teknik Elektro dan Komputer*, 4(2), pp. 18-25.

Mirzaie, A. & Asadollahi, A., 2011. Mobile Commece Beyond Electronik Commerce: Issue and Challenges. *Asian Journal of Business and Management Sciences*, 1(2), pp. 119-129.

Mortana, S. & Noor, M. M., 2010. Pengembangan Customer Relationship Management Berbasis Sistem E-Commerce. *CommIT*, 4(2), pp. 139-149.

Ngatimin, 2013. Perancangan Aplikasi E-Commerce Toko Buku Qisthi Dengan Menggunakan Metode Market Basket Analysis. *Pelita Informatika Budi Darma*, V(1), pp. 26-31.

Niranjanamurthy, Kavyashree, S. Jagannath & Chahar, D., 2013. Analysis of E-Commerce and M-Commerce: Advantages, Limitations, and Security Issues. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(6), pp. 2361-2370.

Nursalima, F. I., Tolle, H. & Arwani, I., 2014. RANCANG BANGUN APLIKASI MOBILE PEDOMAN DAN CATATAN IBU DAN ANAK. *Repositori Jurnal Mahasiswa PTIIK UB*, 3(6), pp. 1-9.

Pramudiya, H. E., Hardarkho, Y. D. & Rahayu, F. S., 2015. Pengimplementasian CRM Pada Pembangunan E-Commerce untuk Usaha Mikro Kecil Menengah (Studi Kasus: Dolanan Puzzle). *Jurnal Buana Informatika*, 6(4), pp. 257 - 268.

Sampurna, O. T., Handojo, A. & Noertjahyana, A., 2013. Pembuatan Aplikasi Sales berbasis Mobile Device

menggunakan PhoneGap. *Jurnal Infra*, 3(2), pp. 89 - 95.

Scornavacca & Barnes, 2006. Barcode enable m-commerce: strategic implications and business models. *International Journal of Mobile Communications*, 4(2), pp. 163-177.

Suarga, H., Amil, I. A. & Mansur, 2013. *TEKNOLOGI MOBILE COMMERCE AUTO REPLY UNTUK KONSEP BUSENESS TO COSTUMERS*, Makassar: Program Pascasarjana UNHAS.

Utami, S. S., 2010. PENGARUH TEKNOLOGI INFORMASI DALAM. *Jurnal Akuntansi dan Sistem Teknologi Informasi*, 8(1), p. 61 - 67.

Wijaya, B. D., E.M.A, F. & Fiade, A., t.thn. Implementasi JSON Parsing Pada Aplikasi Mobile E-Commerce.

Wirdasari, D., 2009. TEKNOLOGI E-COMMERCE DALAM PROSES BISNIS. *Jurnal SAINTIKOM*, 7(2), pp. 324 - 335.

Xenix, P.S., 2015. *Pembangunan Aplikasi Jual Beli Online Toko Koen-B Fashion Berbasis Mobile*. Skripsi. Yogyakarta: Universitas Atma Jaya Yogyakarta Universitas Atma Jaya Yogyakarta.

Yingseng Du & Youchun Tang, 2014. Study on the Development of O2O E-commerce Platform of China from the Perspective of Offline Service Quality. *International Journal of Business and Social Science*, 5(4), pp. 308-312.

Yolanda, E., 2013. *Perancangan Aplikasi E-Commerce Berbasis Web Pada Pempek Rendy*. [Seni] (Program Studi Sistem Informasi Fakultas Teknik dan Ilmu Komputer Universitas Komputer Indonesia).

SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

MEPAMS

(Mobile Commerce "Pamella Swalayan")

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Yakobos Agung Yudha Pratama / 12. 07. 06917

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		SKPL-MEPAMS		1/50
		Revisi		

Program Studi Teknik Informatika	SKPL -APJOLI	1/ 43
----------------------------------	--------------	-------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

DAFTAR ISI

1	Pendahuluan	6
1.1	Tujuan	6
1.2	Lingkup Masalah	6
1.3	Definisi, Akronim dan Singkatan	7
1.4	Referensi	8
1.5	Deskripsi umum (Overview)	8
2	Deskripsi Kebutuhan	9
2.1	Perspektif produk	9
2.2	Fungsi Produk	11
2.3	Karakteristik Pengguna	17
2.4	Batasan-batasan	17
2.5	Asumsi dan Ketergantungan	18
3	Kebutuhan khusus	18
3.1	Kebutuhan antarmuka eksternal	18
4	Spesifikasi Rinci Kebutuhan	22
4.1	Spesifikasi Kebutuhan Fungsionalitas	22

DAFTAR GAMBAR

Gambar 1	Arsitektur Perangkat Lunak MEPAMS.....	11
Gambar 2	Use Case Diagram Untuk Mobile.....	22
Gambar 3	Use Case Diagram Untuk Web.....	23



1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak MEPAMS (Mobile Commerce "Pamella Swalayan") untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna) dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-MEPAMS ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak MEPAMS dikembangkan dengan tujuan untuk:

1. Mengelola proses pemesanan produk oleh pelanggan
2. Menangani pengelolaan data produk
3. Menangani pencarian data produk
4. Melayani pengelolaan data pelanggan.
5. Memberikan layanan *history* transaksi, komentar dan berbagi info produk.
6. Memberikan layanan rekomendasi produk

Dan berjalan pada lingkungan dengan *platform* Android untuk aplikasi *mobile* dan untuk aplikasi *web* dapat berjalan dengan *web browser*.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-MEPAMS-XXX	Kode yang merepresentasikan kebutuhan pada MEPAMS (Pamella Swalayan) dimana XXX merupakan nomor fungsi produk.
MEPAMS	Perangkat lunak pengelolaan pada <i>mobile</i> dan <i>web</i> .
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>network global</i> yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
Pamella Swalayan	Pamella Swalayan merupakan sebuah toko swalayan yang berada di Yogyakarta.

Member	Pelanggan dari Pamela Swalayan yang bertransaksi di swalayan melalui aplikasi
Mobile Commerce	Merupakan proses pembelian ataupun penjualan dari jasa maupun barang melalui jaringan internet atau <i>online</i> berbasis aplikasi <i>mobile</i>
Hybrid Mobile Application	Perangkat lunak mobile aplikasi yang dibangun menggunakan bahasa pemrograman <i>HTML 5</i> dan <i>Javascript</i> .
Market Basket Analysis	Metode yang dipakai dalam menganalisis keranjang belanja untuk menentukan keterkaitan barang yang dibeli pelanggan menggunakan <i>association rules</i> dan algoritma apriori

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Xenix Putra Sasongko, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) APJOLI*, Universitas Atma Jaya Yogyakarta.

1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas tiga bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak MEPAMS yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak MEPAMS tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak MEPAMS yang akan dikembangkan.

2 Deskripsi Kebutuhan

2.1 Perspektif produk

MEPAMS merupakan perangkat lunak yang dibangun untuk meningkatkan layanan penjualan dan promosi produk menggunakan mobile dengan didukung pengelolaan produk di web. Perangkat lunak ini membantu Pamela Swalayan dalam membangun sebuah inovasi layanan yang dapat menangani proses jual beli produk yang terjadi di Pamela Swalayan. Ditambah dengan penerapan *Market Basket Analysis* untuk menganalisis keterkaitan barang yang dibeli oleh pelanggan Pamela Swalayan kemudian dihasilkan sebuah rekomendasi produk bagi pelanggan. Layanan ini selanjutnya dapat dipergunakan oleh Pamela Swalayan sendiri maupun dipergunakan oleh pelanggan dari Pamela Swalayan. Perangkat lunak ini dibagi menjadi dua bagian yaitu aplikasi mobile dan aplikasi web.

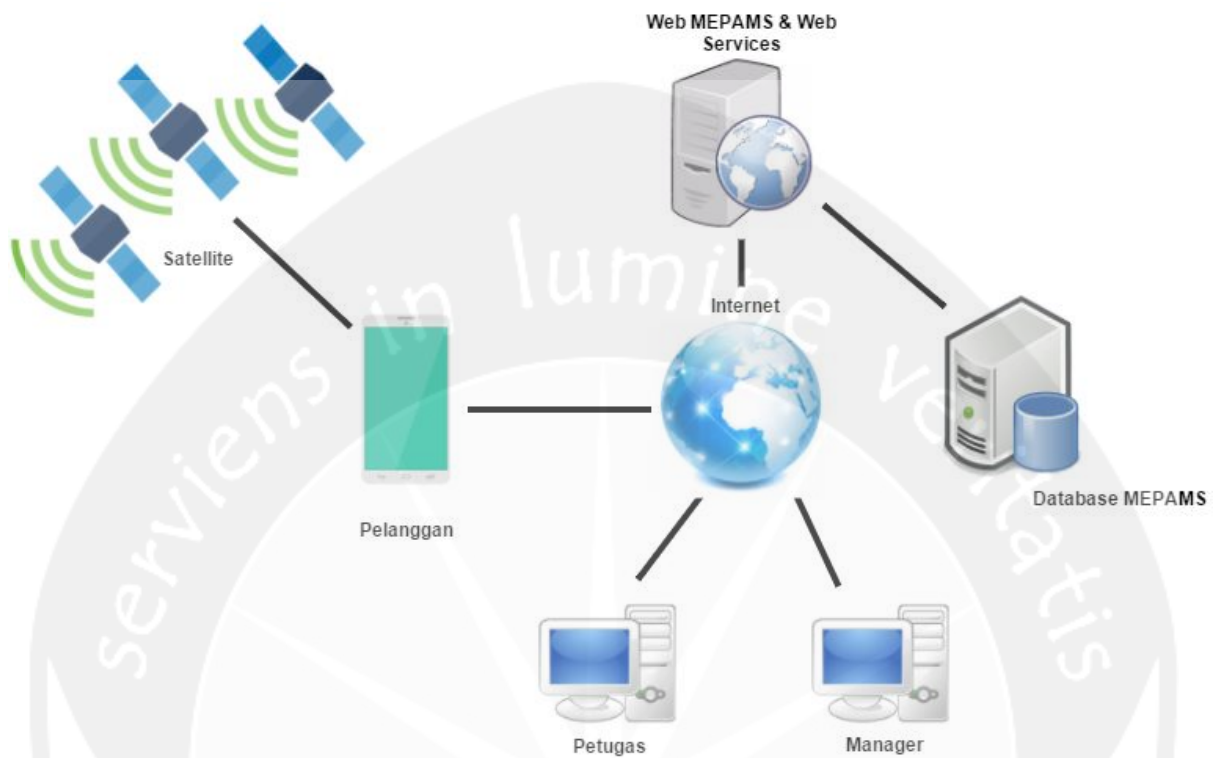
Pada aplikasi *mobile* digunakan oleh pihak pelanggan. Aplikasi *mobile* ini menangani antara lain: pengelolaan data pelanggan, menampilkan produk,

melakukan pemesanan, melihat *history* pemesanan, memberi komentar produk, berbagi info produk oleh pelanggan. Untuk aplikasi web, sistem ini digunakan hanya untuk administrator / pihak Pamela Swalayan. Melalui aplikasi web tersebut pihak Pamela Swalayan dapat mengelola data barang, mengelola informasi lokasi pamella, mengelola pemesanan yang dilakukan oleh pelanggan, mengelola laporan pemesanan, mengelola saran/keluhan.

Perangkat lunak MEPAMS ini berjalan pada *mobile device*, dan dibuat menggunakan bahasa pemrograman *HTML5*, *CSS*, dan *Javascript*. Untuk lingkungan pemrograman menggunakan *Ionic Framework* dan *AngularJS*. Sedangkan untuk *database*, akan dipergunakan *MySQL*, dan kedua aplikasi, *mobile* dan *web* akan mengacu kepada *database* yang sama.

Pelanggan akan berinteraksi dengan aplikasi melalui antarmuka GUI(*Graphical User Interface*). Pada aplikasi seperti terlihat pada Gambar 1., arsitektur perangkat lunak yang digunakan berupa *client server*, dimana data disimpan di *server*. Pelanggan dapat mengakses data yang ada di *server* tersebut secara *online* dengan memanggil *web service* pada *website* yang tersedia di *web server*.

Inputan data yang dimasukkan akan disimpan dalam *database server*, sehingga jika ada pencarian data, maka data yang diinginkan akan dicari ke *database server* yang selanjutnya dikirimkan ke *client* yang meminta melalui *web server*.



Gambar 1. Arsitektur Perangkat lunak MEPAMS

2.2 Fungsi Produk

Fungsi produk perangkat lunak MEPAMS adalah sebagai berikut :

APLIKASI MOBILE

1. Fungsi Login (**SKPL-MEPAMS-001**) .

Merupakan fungsi yang digunakan oleh semua pelanggan untuk dapat masuk dalam aplikasi.

2. Fungsi Mendaftar/Registrasi (**SKPL-MEPAMS-002**) .

Merupakan fungsi yang digunakan untuk mendaftarkan diri sebagai pelanggan pada aplikasi MEPAMS.

3. Fungsi Pengelolaan Data Pelanggan (**SKPL-MEPAMS-003**) .

Merupakan fungsi yang digunakan oleh semua pelanggan yang telah terdaftar untuk mengelola data pelanggan.

Fungsi *Pengelolaan Data Pelanggan* mencakup :

a. Fungsi *Ubah Password* (**SKPL-MEPAMS-003-01**) .

Merupakan fungsi yang digunakan pelanggan untuk merubah *password*.

b. Fungsi *Edit Data Pelanggan* (**SKPL-MEPAMS-003-02**) .

Merupakan fungsi yang digunakan untuk mengubah data-data pelanggan.

4. Fungsi Cari Data Produk (**SKPL-MEPAMS-004**) .

Merupakan fungsi yang digunakan untuk mencari data produk berdasarkan kata kunci yang digunakan.

5. Fungsi Tampil Data Produk (**SKPL-MEPAMS-005**) .

Merupakan fungsi yang digunakan untuk menampilkan data produk, dan terbagi menjadi:

a. Tampil Rekomendasi Produk (**SKPL-MEPAMS-005-01**) .

Merupakan fungsi yang digunakan untuk menampilkan rekomendasi produk.

b. Tampil Promo Produk (**SKPL-MEPAMS-005-02**) .

Merupakan fungsi yang digunakan untuk menampilkan promo produk.

c. Tampil Katalog Produk **(SKPL-MEPAMS-005-03)** .

Merupakan fungsi yang digunakan untuk menampilkan data produk berdasarkan katalog.

d. Tampil Produk Terlaris **(SKRL-MEPAMS-005-04)** .

Merupakan fungsi yang digunakan untuk menampilkan data produk yang laris terjual kepada pelanggan

6. Fungsi *Comment* dan *Share* **(SKPL-MEPAMS-006)** .

Merupakan fungsi yang digunakan oleh semua pelanggan yang telah terdaftar untuk memberi komentar atas produk dan berbagi informasi produk.

7. Fungsi Pemesanan **(SKPL-MEPAMS-007)**

Merupakan fungsi yang digunakan oleh semua pelanggan yang telah terdaftar untuk melakukan pemesanan, dan terdiri atas:

a. Fungsi Keranjang Belanja **(SKPL-MEPAMS-007-01)** .

Merupakan fungsi yang digunakan menampung produk yang dipilih oleh pelanggan.

b. Fungsi Kelola Keranjang Belanja **(SKPL-MEPAMS-007-02)** .

Merupakan fungsi digunakan untuk mengelola keranjang belanja yang berisi produk.

c. Fungsi Upload Pemesanan **(SKPL-MEPAMS-007-03)** .

Merupakan fungsi yang digunakan untuk mengirimkan data pemesanan produk dari keranjang belanja ke *database server*.

8. Fungsi *History Transaksi* (**SKPL-MEPAMS-008**) .

Merupakan fungsi yang digunakan untuk menampilkan history transaksi kepada pelanggan.

9. Fungsi Formulir Konfirmasi Pembayaran (**SKPL-MEPAMS-009**) .

Merupakan fungsi yang digunakan oleh pelanggan untuk mengirimkan konfirmasi pembayaran dari transaksi yang dilakukan.

APLIKASI WEB

10. Fungsi *Login Web* (**SKPL-MEPAMS-010**)

Merupakan fungsi yang digunakan oleh petugas untuk bisa masuk ke dalam pengelolaan web.

a. Fungsi *Reset Password* (**SKPL-MEPAMS-010-01**) .

Merupakan fungsi yang digunakan petugas untuk *reset password*.

11. Fungsi Pengelolaan Transaksi (**SKPL-MEPAMS-011**)

Merupakan fungsi yang digunakan oleh petugas untuk mengelola transaksi pemesanan yang terdiri atas:

a. Fungsi *Update Status Pembayaran* (**SKPL-MEPAMS-011-001**)

Merupakan fungsi yang digunakan petugas untuk mengubah status pembayaran yang dilakukan oleh pelanggan.

b. Fungsi *Update Status Barang* (**SKPL-MEPAMS-011-003**)

Merupakan fungsi yang digunakan petugas untuk mengubah status barang yang dipesan oleh pelanggan.

12. Fungsi *Pengelolaan Data* Petugas (**SKPL-MEPAMS-012**) .

Merupakan fungsi yang dipergunakan oleh Manager untuk mengelola data Petugas.

Fungsi *Pengelolaan Data Petugas* mencakup :

a. Fungsi Entri Data Petugas (**SKPL-MEPAMS-012-01**) .

Merupakan fungsi yang digunakan untuk menambahkan data petugas.

b. Fungsi Hapus Data Petugas (**SKPL-MEPAMS-012-02**) .

Merupakan fungsi yang digunakan untuk menghapus data petugas.

c. Fungsi Edit Data Petugas (**SKPL-MEPAMS-012-03**) .

Merupakan fungsi yang digunakan untuk mengedit data petugas.

d. Fungsi Tampil Data Petugas (**SKPL-MEPAMS-012-04**) .

Merupakan fungsi yang digunakan untuk menampilkan data petugas.

13. Fungsi Lihat Data Pelanggan (**SKPL-MEPAMS-013**) .

Merupakan fungsi untuk melihat data pelanggan dan pemesanannya.

14. Fungsi Pengelolaan Produk (**SKPL-MEPAMS-014**) .

Merupakan fungsi yang dipergunakan oleh petugas untuk mengelola data Produk.

a. Fungsi Entri Produk (**SKPL-MEPAMS-014-01**) .

Merupakan fungsi yang digunakan oleh petugas untuk memasukkan data Produk.

b. Fungsi Ubah Produk (**SKPL-MEPAMS-014-02**) .

Merupakan fungsi yang digunakan oleh petugas untuk mengubah data Produk.

c. Fungsi Hapus Produk (**SKPL-MEPAMS-014-03**) .

Merupakan fungsi yang digunakan oleh petugas untuk menghapus data Produk.

15. Fungsi Pengelolaan Saran/keluhan(**SKPL-MEPAMS-015**) .

Merupakan fungsi yang dipergunakan oleh Petugas untuk mengelola data saran/keluhan.

a. Fungsi Tampil Saran/keluhan (**SKPL-MEPAMS-015-01**) .

Merupakan fungsi yang digunakan oleh petugas untuk menampilkan data saran/keluhan.

b. Fungsi Balas Saran/keluhan (**SKPL-MEPAMS-015-02**)

Merupakan fungsi yang digunakan oleh petugas untuk membalas saran/keluhan

c. Fungsi Hapus Saran/keluhan (**SKPL-MEPAMS-015-03**) .

Merupakan fungsi yang digunakan oleh petugas untuk menghapus data saran/keluhan.

16. Fungsi Cetak Laporan(**SKPL-MEPAMS-016**) .

Fungsi yang digunakan oleh manager untuk mencetak laporan Pemesanan yang dilakukan oleh pelanggan dalam rentan waktu harian, mingguan, dan bulanan.

17. Fungsi Pengelolaan Location(**SKPL-MEPAMS-017**) .

Fungsi yang digunakan oleh petugas untuk melakukan *update* terhadap *location* Pamella Swalayan yang tampil di aplikasi *mobile*

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak MEPAMS adalah sebagai berikut :

1. Memahami pengoperasian perangkat *device/mobile*.
2. Memahami penggunaan *Aplikasi Mobile e-Commerce* Pamella Swalayan(MEPAMS) .

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak MEPAMS tersebut adalah :

1. Kebijakanaksanaan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak MEPAMS.

2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan) .

Program Studi Teknik Informatika	SKPL – MEPAMS	17/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada perangkat Mobile Smartphone yang menggunakan system operasi versi android 4.0 ke atas, serta aplikasi web browser apapun.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antar muka eksternal pada perangkat lunak MEPAMS meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form yang ada pada layar *mobile device*.

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak MEPAMS adalah:

1. Perangkat *mobile* Android minimal 4.1 Android *version*.
2. Perangkat *personal computer*(PC)/Laptop minimum Windows 7.

3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak MEPAMS adalah sebagai berikut :

Program Studi Teknik Informatika	SKPL – MEPAMS	18/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Nama : IE/Firefox/Chrome/Opera/dll
Sumber : berbagai sumber
Sebagai aplikasi web browser untuk membuka pengelolaan web.
2. Nama : Android System Operation
Sumber : berbagai sumber
Sebagai sistem operasi yang berjalan di handphone/smartphone.
3. Nama : Windows 10
Sumber : Microsoft
Sebagai sistem operasi yang digunakan dalam membangun aplikasi MEPAMS
4. Nama : NodeJS CLI
Sumber : Ionic Framework
Sebagai user interface untuk membuat perintah dasar pada pembangunan aplikasi ionic framework.
5. Nama : Sublime
Sumber : berbagai sumber
Sebagai text editor dalam membangun aplikasi menggunakan ionic framework dan angularJS.
6. Nama : MySQL
Sumber : MySQL
Sebagai database management sistem (DBMS) yang digunakan untuk menyimpan data di sisi server.
- 7 Nama : Android SDK
Sumber : Google Inc.
Sebagai Android library dalam perancangan aplikasi MEPAMS.

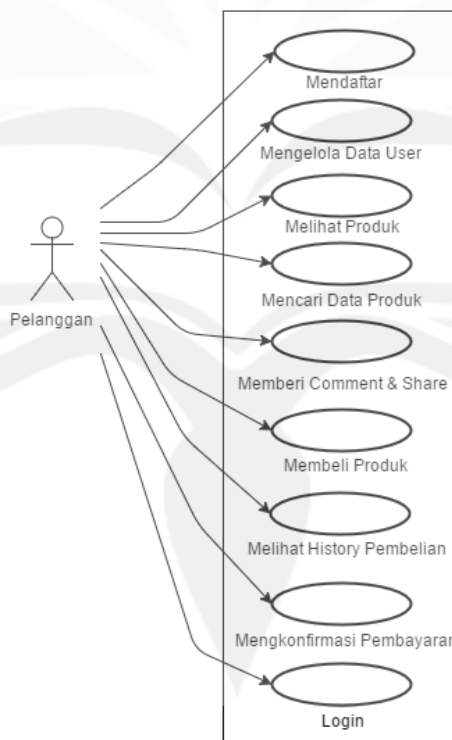
3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak MEPAMS menggunakan protocol HTTP.

3.2 Kebutuhan fungsionalitas Perangkat Lunak

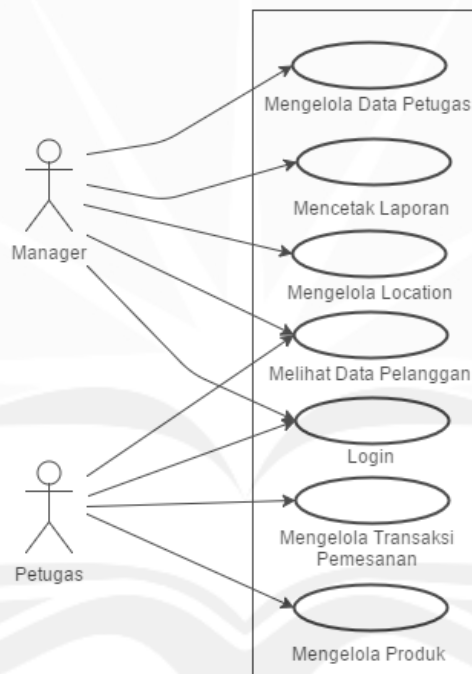
3.2.1 Use Case Diagram

Gambar 2. menunjukkan pembagian use case untuk aplikasi mobile yang akan dibangun dan akan digunakan oleh pelanggan Pamela Swalayan. Pada use case tersebut pelanggan dapat mendaftarkan diri sebagai pelanggan, login, mengelola data pribadi, melihat produk, mencari produk, memberi komentar dan membagi produk melalui media social, membeli produk, melihat history pemesanan, dan melakukan konfirmasi pembayaran.



Gambar 2. Use Case Diagram Untuk Mobile

Gambar 3. menunjukkan pembagian use case untuk aplikasi web yang akan dibangun untuk pihak Pamela Swalayan. Pada use case terdapat dua aktor yaitu manager, dan petugas. Manager dapat login, melihat data transaksi pelanggan, mencetak laporan, mengelola informasi lokasi Pamela Swalayan, dan melihat hasil market basket analisis. Kemudian petugas dapat melakukan login, mengelola transaksi pemesanan dari pelanggan, dan mengelola produk.



Gambar 3. Use Case Diagram Untuk Web

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

APLIKASI MOBILE

4.1.1 Use Case Specification : Login

1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke aplikasi mobile. Login didasarkan pada sebuah *username* dari pelanggan dan *password* yang berupa rangkaian karakter.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login.
2. Aplikasi menampilkan antarmuka untuk login.
3. Aktor memasukkan *username* dan *password*.
4. Aplikasi memeriksa *username* dan *password* yang diinputkan aktor.

E-1 *Password* atau *username* tidak sesuai.

5. Aplikasi memberikan akses ke aktor.
6. Use Case ini selesai.

5. Alternative Flow

None

6. Error Flow

E-1 *Password* atau *username* tidak sesuai.

1. Aplikasi menampilkan peringatan bahwa *username* atau *password* tidak sesuai.

2. Aplikasi menawarkan fungsi forget password kepada aktor.
3. Aktor memasukkan email yang di registrasikan di aplikasi.
4. Aplikasi mengirimkan email reset password.
5. Aktor membuat password baru.
6. Kembali ke *Basic Flow* langkah ke 3.

7. PreConditions

None

8. PostConditions

1. Aktor memasuki aplikasi dan dapat menggunakan fungsi-fungsi pada aplikasi.

4.1.2 Use Case Spesification : Registrasi/Daftar

1. Brief Description

Use Case ini digunakan oleh aktor untuk mendaftarkan data diri yang akan disimpan di *database* oleh aplikasi.

2. Primary Actor

2. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan registrasi.
2. Aplikasi menampilkan antarmuka untuk memasukkan data diri pelanggan.
3. Aktor menginputkan data pelanggan.
4. Aktor meminta aplikasi untuk menyimpan data yang telah diinputkan.

5. Aplikasi mengecek data pelanggan yang telah diinputkan.

E-1 Data Pelanggan yang diinputkan aktor salah.

6. Aplikasi menyimpan data pelanggan ke database.

7. *Use Case* selesai.

5. Alternative Flow

None

6. Error Flow

E-1 Terdapat kolom yang tidak sesuai.

1. Sistem menampilkan peringatan bahwa terdapat kolom yang tidak sesuai.
2. Kembali ke *Basic Flow* langkah ke 4.

7. PreConditions

None

1. PostConditions

1. Data pelanggan di database telah ter-update.

4.1.3 Use Case Spesification : Pengelolaan data pelanggan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data dari pelanggan. Aktor dapat melakukan edit data pelanggan (selain *username*), mengubah *password*.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pelanggan.
2. Aplikasi menampilkan antarmuka pengelolaan data pelanggan.
3. Aplikasi menampilkan data profil pelanggan (*password* tidak ditampilkan).
4. Sistem memberikan pilihan untuk melakukan edit data pelanggan (selain *username*), atau mengubah *password*.
5. Aktor memilih untuk melakukan edit data pelanggan.
 - A-1 Aktor memilih untuk melakukan ubah *password*
6. Aktor mengubah data pelanggan.
7. Aktor meminta aplikasi untuk menyimpan data pelanggan yang telah diinputkan.
8. Aplikasi mengecek data pelanggan yang telah diinputkan.
 - E-1 Data anggota yang diinputkan aktor salah.
9. Aplikasi menyimpan data pelanggan ke *database*.
10. Use Case selesai.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan ubah *password* pelanggan.
 1. Sistem menampilkan antar muka ubah *password*.
 2. Aktor mengisi kolom yang sudah disediakan.

3. Aktor meminta aplikasi untuk menyimpan *password* pelanggan yang telah diedit.

4. Sistem melakukan pengecekan terhadap *password* pelanggan yang telah diedit.

E-2 Data pelanggan yang telah diedit salah.

5. Sistem menyimpan *password* pelanggan yang telah diedit ke *database*.

6. Berlanjut ke *Basic Flow* langkah ke 10.

6. Error Flow

E-1 Data pelanggan yang diinputkan aktor salah.

1. Aplikasi memberikan pesan peringatan bahwa data yang diinputkan salah.

2. Kembali ke *Basic Flow* Langkah ke 4.

E-2 *Password* pelanggan yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa *password* yang diedit salah.

2. Kembali ke *Alternative Flow A-1* Langkah ke 2.

7. PreConditions

1. *Use Case Login* telah dilakukan.

2. Aktor telah memasuki sistem.

8. PostConditions

1. Data pelanggan di *database* telah terupdate.

4.1.4 Use Case Spesification : Cari Data Produk

1. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan pencarian data produk berdasarkan nama produk.

Program Studi Teknik Informatika	SKPL – MEPAMS	26/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

1. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan pencarian data produk.
2. Aplikasi Memberikan SearchBox untuk inputan.
3. Aktor mengetikkan nama produk yang ingin dicari.
4. Aplikasi menampilkan data produk berdasarkan nama yang diinputkan aktor.
E-1 Nama produk yang dicari tidak ada.
5. *Use Case* selesai.

5. Alternative Flow

None

6. Error Flow

- E-1 Nama produk yang dicari oleh actor tidak ada
- a. Aplikasi memberikan pesan peringatan bahwa produk yang dicari tidak ada.
 - b. Kembali ke *Basic Flow* Langkah ke 3.

7. PreConditions

1. *Use Case* Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Aktor mendapatkan data produk yang dicari.

4.1.5 Use Case Spesification : Fungsi Comment and Share

1. Brief Description

Use Case ini digunakan aktor untuk melakukan comment dan share terhadap produk. Aktor dapat meng-entry komentar dan aktor dapat membagikan informasi produk yang dipilih lalu di share ke media sosial yang dipilih.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan *comment* and *share*.
2. Aplikasi menampilkan antarmuka *comment* and *share*.
3. Aplikasi memberi pilihan kepada aktor untuk memilih *Rate* atau *Share*.
4. Aktor memilih untuk melakukan entri *comment*.
A-1 Aktor memilih untuk melakukan *share*.
5. Aktor menginputkan data *comment* (Judul komentar, komentar).
6. Sistem menyimpan data *comment* ke *database*.
7. Use Case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan *share*.

1. Aplikasi menampilkan pilihan untuk memilih bagian informasi produk ke aplikasi media sosial yang dipunyai aktor.
2. Aktor memilih produk dan membagikan ke media sosial yang telah dipilih pelanggan.

Program Studi Teknik Informatika	SKPL – MEPAMS	28/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3. Berlanjut ke *Basic Flow* langkah ke 8.

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data *comment* di *database* telah terupdate

4.1.6 Use Case Spesification : Fungsi Tampil Produk

1. Brief Description

Use Case ini digunakan aktor untuk melihat produk yang dikategorikan ke dalam produk promo dan produk katalog.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk membuka aplikasi
2. Aplikasi menampilkan antarmuka aplikasi berupa produk promo dan produk yang dikategorikan ke dalam katalog
3. Use Case selesai.

5. Alternative Flow

None

6. Error Flow

None

7. PreConditions

None

8. PostConditions

1. Aktor berhasil masuk ke dalam sistem dan melihat produk

4.1.7 Use Case Spesification : Pemesanan Produk

1. Brief Description

Use Case ini digunakan aktor untuk melakukan pemesanan produk yang ada di aplikasi. Aktor dapat memilih produk, menghapus pesanan produk, mengubah jumlah pemsanan, dan upload pemesanan.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih produk.
2. Aplikasi menampilkan antarmuka produk.
3. Aplikasi memberi pilihan kepada aktor membeli atau menambah barang ke keranjang belanja.
A-1 Aktor memilih menambah barang ke keranjang belanja.
4. Aplikasi menampilkan produk yang dibeli didalam keranjang belanja.
A-2 Aktor memilih mengubah jumlah pesanan.
A-3 Aktor memilih menghapus pesanan.
5. Aktor memilih melanjutkan proses pemesanan.
6. Aplikasi menampilkan informasi pilihan pengambilan barang, informasi pembayaran, dan penagihan.
7. Aktor memasukkan pilihan pengambilan barang.

8. Aplikasi menampilkan pilihan lanjut proses verifikasi pemesanan
9. Aktor memilih melanjutkan proses verifikasi pemesanan.
10. Aplikasi menampilkan informasi detail pemesanan dan pilihan upload transaksi.
11. Aktor memilih upload transaksi.
12. Aplikasi menyimpan data pemesanan di database.
13. Use case selesai

5. Alternative Flow

A-1 Aktor memilih menambah barang ke keranjang

1. Aplikasi menambahkan produk ke keranjang belanja.
2. Berlanjut ke *Basic Flow* langkah ke 4.

A-2 Aktor memilih mengubah jumlah pesanan.

1. Aplikasi menampilkan pilihan jumlah pesanan.
2. Aktor memilih jumlah pesanan.
3. Berlanjut ke *Basic Flow* langkah ke 5.

A-3 Aktor memilih menghapus pesanan.

1. Aplikasi menampilkan pilihan hapus atau batal.
2. Aktor memilih hapus barang.
3. Aplikasi menghapus barang dari keranjang belanja
4. Berlanjut ke *Basic Flow* langkah ke 5.

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

Program Studi Teknik Informatika	SKPL – MEPAMS	31/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. PostConditions

1. Data kelola pemesanan pelanggan di *database* telah tersimpan

4.1.8 Use Case Spesification : Melihat History Transaksi

1. Brief Description

Use Case ini digunakan oleh aktor untuk melihat history transaksi dari transaksi yang sudah dilakukan.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aplikasi menampilkan pilihan history transaksi
2. Aktor memilih history transaksi
3. Aplikasi menampilkan history transaksi
 - A-1 Aplikasi menampilkan pilihan melihat detail barang dari transaksi.
 - A-2 Aplikasi menampilkan pilihan hapus transaksi.
4. Use case selesai

5. Alternative Flow

A-1 Aplikasi menampilkan pilihan melihat detail barang dari transaksi.

1. Aktor memilih lihat detail transaksi.
2. Aplikasi menampilkan detail transaksi.
3. Berlanjut ke *Basic Flow* langkah ke 4.

A-2 Aplikasi menampilkan pilihan hapus transaksi.

1. Aktor memilih hapus transaksi.

2. Aplikasi menghapus transaksi.
3. Berlanjut ke *Basic Flow* langkah ke 4.

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan
2. Aktor telah memasuki sistem

8. PostConditions

1. *History* transaksi berhasil di tampilkan.

4.1.9 Use Case Spesification : Konfirmasi Pembayaran

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengkonfirmasi bentuk pembayaran yang dilakukan menggunakan transfer maupun bayar langsung ditempat.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih mengkonfirmasi pembayaran
2. Aplikasi menampilkan kolom konfirmasi pembayaran
3. Aktor mengisi kolom konfirmasi pembayaran.
4. Use Case selesai.

5. Alternative Flow

None

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan

2. Aktor telah memasuki sistem

8. PostConditions

1. Konfirmasi pembayaran berhasil disimpan di *database*.

APLIKASI WEB

4.1.10 Use Case Spesification : Login Web

1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah *username* dari pelanggan dan *password* yang berupa rangkaian karakter.

2. Primary Actor

1. Petugas

3. Supporting Actor

None

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan login.
2. Sistem menampilkan antarmuka untuk login.
3. Aktor memasukkan *username* dan *password*.
4. Sistem memeriksa *username* dan *password* yang diinputkan aktor.
E-1 *Password* atau *username* tidak sesuai.
5. Sistem memberikan akses ke aktor.
6. *Use Case* ini selesai.

5. Alternative Flow

None

6. Error Flow

E-1 *Password* atau *username* tidak sesuai.

Program Studi Teknik Informatika	SKPL – MEPAMS	34/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem menampilkan peringatan bahwa *username* atau *password* tidak sesuai.
2. Aplikasi menawarkan fungsi *forget password* kepada aktor.
3. Aktor memasukkan email yang di registrasikan di aplikasi.
4. Aplikasi mengirimkan email reset password.
5. Aktor membuat password baru.
6. Kembali ke *Basic Flow* langkah ke 3.

7. PreConditions

None

8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

4.1.11 Use Case Spesification : Kelola Transaksi

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola transaksi yang dilakukan oleh pelanggan.

2. Primary Actor

1. Petugas

3. Supporting Actor

None

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan pengelolaan transaksi
2. Sistem menampilkan antarmuka pengelolaan transaksi.
 - A-1 Sistem menampilkan pilihan update transaksi
 - A-2 Sistem menampilkan pilihan lihat detail transaksi

Program Studi Teknik Informatika	SKPL – MEPAMS	35/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3. *Use Case* ini selesai.

5. Alternative Flow

A-1 Sistem menampilkan pilihan update transaksi

1. Aktor memilih update transaksi
2. Sistem menampilkan pilihan mengubah status bayar dan status barang.
3. Aktor menyimpan perubahan status.
4. Kembali ke *Basic Flow* langkah ke 3.

A-2 Sistem menampilkan pilihan lihat detail transaksi.

1. Aktor memilih lihat detail transaksi
2. Sistem menampilkan detail transaksi
3. Kembali ke *Basic Flow* langkah ke 3.

6. Error Flow

None

7. PreConditions

1. *Use Case* login telah dilakukan
2. Aktor telah memasuki sistem

8. PostConditions

1. Sistem menyimpan perubahan status transaksi.

4.1.12 Use Case Spesification : Pengelolaan data petugas

1. Brief Description

Use Case ini digunakan aktor untuk mencatat dan mengelola data petugas. Aktor dapat melakukan entri data petugas, edit data petugas, dan hapus data petugas.

2. Primary Actor

1. Manager

3. Supporting Actor

None

Program Studi Teknik Informatika	SKPL – MEPAMS	36/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4. **Basic Flow**

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan pencatatan data petugas baru.
2. Sistem menampilkan antarmuka pencatatan data petugas.
3. Aktor memilih untuk melakukan entri data petugas.
A-1 Aktor memilih untuk melakukan edit data petugas.
4. Aktor menginputkan data petugas.
5. Aktor meminta sistem untuk menyimpan data petugas yang telah diinputkan.
6. Sistem mengecek data petugas yang telah diinputkan.
E-1 Data petugas yang diinputkan aktor salah.
7. Sistem menyimpan data petugas ke *database*.
8. *Use Case* selesai.

5. **Alternative Flow**

- A-1 Aktor memilih untuk melakukan edit data petugas.
1. Sistem menampilkan data petugas.
 2. Aktor mengedit data petugas yang sudah ditampilkan.
 3. Aktor meminta sistem untuk menyimpan data petugas yang telah diedit.
 4. Sistem melakukan pengecekan terhadap data petugas yang telah diedit.
E-2 Data petugas yang telah diedit salah.
 5. Berlanjut ke *Basic Flow* langkah ke 8.

6. **Error Flow**

None

7. PreConditions

1. Use Case login telah dilakukan
2. Aktor telah memasuki sistem

8. PostConditions

1. Data petugas telah terupdate di database.

4.1.15 Use Case Spesification : Pengelolaan data produk

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data produk. Aktor dapat melakukan entri produk baru, edit data produk, hapus data produk.

2. Primary Actor

1. Petugas

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data produk.
2. Sistem menampilkan antarmuka pengelolaan data produk.
3. Sistem menampilkan data semua produk.
4. Sistem memberikan pilihan untuk melakukan entri new produk, edit data produk, hapus data produk.
5. Aktor memilih untuk melakukan entri produk baru.
 - A-1 Aktor memilih untuk melakukan edit data produk.
 - A-2 Aktor memilih untuk melakukan hapus data produk.
6. Aktor menginputkan data produk.

7. Aktor meminta sistem untuk menyimpan data produk yang telah diinputkan.
8. Sistem mengecek data produk yang telah diinputkan.

E-1 Data produk yang diinputkan aktor salah.

9. Sistem menyimpan data produk ke *database*.
10. *Use Case* selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit data produk.

1. Sistem menampilkan produk.
2. Aktor mengedit data produk yang sudah ditampilkan.
3. Aktor meminta sistem untuk menyimpan data produk yang telah diedit.
4. Sistem melakukan pengecekan terhadap data produk yang telah diedit.

E-2 Data produk yang telah diedit salah.

5. Sistem menyimpan data produk yang telah diedit ke *database*.
6. Berlanjut ke *Basic Flow* langkah ke 8.

A-2 Aktor memilih untuk melakukan hapus data produk.

1. Sistem menampilkan data produk yang ingin dihapus.
2. Aktor memilih untuk menghapus data produk.
3. Berlanjut ke *Basic Flow* langkah 8.

6. Error Flow

E-1 Data produk yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah.

2. Kembali ke *Basic Flow* Langkah ke 4.

E-2 Data produk yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa data yang diedit salah.

2. Kembali ke *Alternative Flow A-1* Langkah ke 2.

7. PreConditions

1. *Use Case* Login telah dilakukan.

2. Aktor telah memasuki sistem

8. PostConditions

1. Data produk di *database* telah terupdate.

4.1.17 Use Case Spesification : Lihat Pelanggan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola pelanggan melalui web. Aktor dapat melihat pelanggan, dan *reset password*.

2. Primary Actor

1. Petugas

3. Supporting Actor

None

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan lihat pelanggan.

2. Sistem menampilkan antarmuka lihat pelanggan.

3. Sistem menampilkan data semua pelanggan.

4. Sistem memberikan pilihan untuk melakukan *reset password* pelanggan.

5. Aktor memilih untuk melakukan *reset password* pelanggan.

Program Studi Teknik Informatika	SKPL – MEPAMS	40/ 43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Sistem mengubah password pelanggan menjadi 'userpass'.
7. Sistem menyimpan data pelanggan ke database.
8. Use Case selesai.

5. Alternative Flow

None

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Password pelanggan telah diubah menjadi 'userpass'.

4.1.18 Use Case Spesification : Cetak Laporan

1. Brief Description

Use Case ini digunakan oleh aktor untuk melihat laporan Pemesanan yang dilakukan dalam rentan waktu harian, mingguan, dan bulanan.

2. Primary Actor

1. Manager

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan lihat laporan.
2. Sistem menampilkan antarmuka lihat laporan.
3. Sistem menampilkan data laporan.
4. Sistem memberikan pilihan untuk mencetak laporan harian, mingguan, atau bulanan.
5. Use Case selesai.

5. Alternative Flow

None

6. Error Flow

None

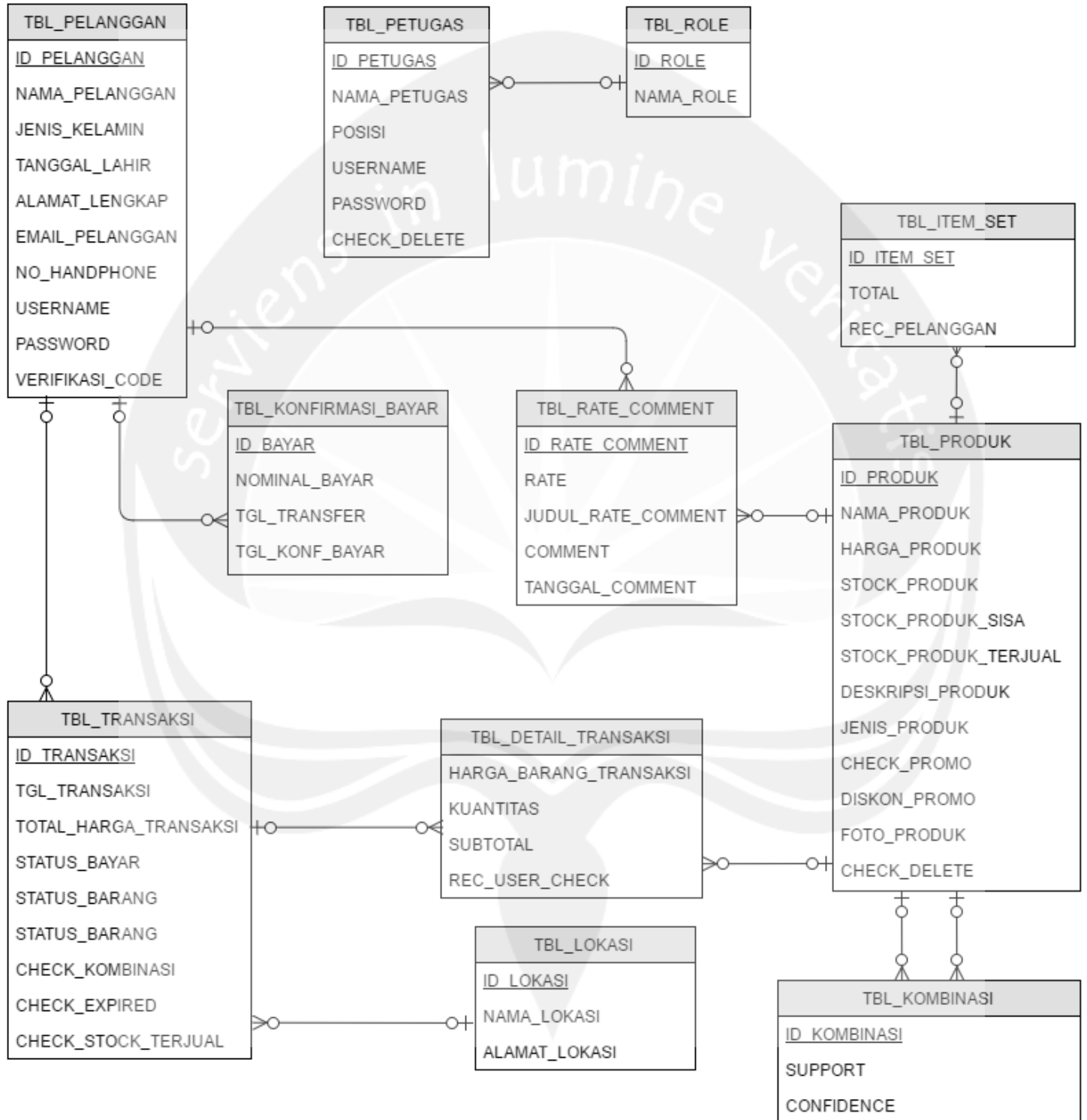
7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Aktor mencetak laporan Pemesanan sesuai yang dipilih.

ERD



DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

MEPAMS

(Mobile Commerce "Pamella Swalayan")

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Yakobos Agung Yudha Pratama / 12. 07. 06917

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		DPPL - MEPAMS		
		Revisi		1 / 70

DAFTAR PERUBAHAN

REVISI	DESKRIPSI
A	
B	
C	
D	
E	
F	

INDEKS TGL	-	A	B	C	D	E	F
DITULIS OLEH							
DIPERIKSA OLEH							
DISETUJUI OLEH							

DAFTAR HALAMAN PERUBAHAN

Halaman	Revisi	Halaman	Revisi

DAFTAR ISI

1. Pendahuluan.....	6
1.1 Tujuan.....	6
1.2 Lingkup Masalah.....	6
1.3 Definisi, Akronim dan Singkatan.....	7
1.4 Referensi.....	8
2 Perancangan Sistem.....	9
2.1 Perancangan Asitektur.....	9
2.2 Perancangan Rinci.....	11
2.2.2 <i>Class Diagram Specific Descriptions</i>	22
2.2.2.29 Specific Design Class Web_LoginUI.....	31
2.2.2.30 Specific Design Class Web_KelolaAkunUI.....	31
2.2.2.31 Specific Design Class Web_KelolaProdukUI.....	32
2.2.2.32 Specific Design Class Web_BuatAkunUI.....	32
2.2.2.33 Specific Design Class Web_TampilPelangganUI.....	32
2.2.2.34 Specific Design Class Web_PengelolaanLaporanUI.....	33
2.2.2.35 Specific Design Class Web_KelolaHelpDeskUI.....	33
2.2.2.36 Specific Design Class Web_KelolaPemesananUI.....	33
2.2.2.37 Specific Design Class Web_KelolaLocationUI.....	34
2.2.2.38 Specific Design Class Web_EmailUI.....	34
2.2.2.39 Specific Design Class Web_Login_Control.....	34
2.2.2.40 Specific Design Class Web_KelolaAkun_Control.....	34
2.2.2.41 Specific Design Class Web_BuatAkun_Control.....	35
2.2.2.42 Specific Design Class Web_KelolaProduk_Control.....	35
2.2.2.43 Specific Design Class Web_KelolaPemesanan_Control.....	35
2.2.2.44 Specific Design Class Web_Location_Control.....	36
2.2.2.45 Specific Design Class Web_Email_Control.....	36
2.2.2.46 Specific Design Class Web_KelolaHelpDeskUI.....	36
2.2.2.47 Specific Design Class Web_TampilPelanggan_Control.....	37
2.2.2.48 Specific Design Class Web_Pengelolaan_Laporan_Control.....	37
2.2.2.49 Specific Design Class Petugas_Model.....	37
2.2.2.50 Specific Design Class Data_Produk_Model.....	38
2.2.2.51 Specific Design Class DataPemesanan_Model.....	39
2.2.2.52 Specific Design Class Pelanggan_Model.....	40
2.2.2.53 Specific Design Class KelolaHelpDesk_Model.....	40
2.2.2.54 Specific Design Class Email_Model.....	41
2.2.2.55 Specific Design Class Location_Model.....	41
3. PERANCANGAN DATA.....	43

DAFTAR GAMBAR

Gambar 2.1 Rancangan Arsitektur MEPAMS Mobile	11
Gambar 2.2 Rancangan Arsitektur MEPAMS Web	12
Gambar 2.2 Sequence Diagram : Login Mobile	13
Gambar 2.3 Sequence Diagram : Mendaftar	13
Gambar 2.4 Sequence Diagram : Pengelolaan Data Pelanggan	14
Gambar 2.5 Sequence Diagram : Fungsi Insert Rate&Share.	14
Gambar 2.6 Sequence Diagram : Fungsi Kelola Pesan	15
Gambar 2.7 Sequence Diagram : Fungsi Kelola Bayar	15
Gambar 2.8 Sequence Diagram : Fungsi Request Geolocation	16
Gambar 2.9 Sequence Diagram : Fungsi Cari Data Produk...	16
Gambar 2.10 Sequence Diagram : Login Web	17
Gambar 2.11 Sequence Diagram : Fungsi Update Status	17
Gambar 2.12 Sequence Diagram : Fungsi Entri Petugas	18
Gambar 2.13 Sequence Diagram : Fungsi Hapus Data Petugas	18
Gambar 2.14 Sequence Diagram : Fungsi Tambah Produk	19
Gambar 2.15 Sequence Diagram : Fungsi Ubah Produk	19
Gambar 2.16 Sequence Diagram : Fungsi Hapus Data Pesan.	20
Gambar 2.17 Sequence Diagram : Fungsi Input Data Pesan.	20
Gambar 2.18 Class Diagram Mobile	21
Gambar 2.19 Class Diagram Web	22
Gambar 3.1 Physical Data Model	49
Gambar 4.1 Rancangan Antarmuka Login Mobile	50
Gambar 4.2 Rancangan Antarmuka Registrasi	51
Gambar 4.3 Rancangan Antarmuka Main Activity	52
Gambar 4.4 Rancangan Antarmuka Kelola Data Pelanggan ..	53
Gambar 4.5 Rancangan Antarmuka Tampil Produk	54
Gambar 4.6 Rancangan Antarmuka Detail Produk	55
Gambar 4.7 Rancangan Antarmuka Review Produk	56
Gambar 4.8 Rancangan Antarmuka Add Review	57
Gambar 4.9 Rancangan Antarmuka Kotak Masuk	58
Gambar 4.9 Rancangan Antarmuka Daftar Pesanan	59
Gambar 4.9 Rancangan Antarmuka Pembayaran	60
Gambar 4.9 Rancangan Antarmuka metode transfer	61
Gambar 4.9 Rancangan Antarmuka Login Web Online	63
Gambar 4.9 Rancangan Antarmuka Buat Admin	64
Gambar 4.9 Rancangan Antarmuka Lihat Admin	65
Gambar 4.9 Rancangan Antarmuka Tabel Belum Bayar	66
Gambar 4.9 Rancangan Antarmuka Tabel Sudah Bayar	67
Gambar 4.9 Rancangan Antarmuka Tabel Buat Pesan	68
Gambar 4.9 Rancangan Antarmuka Tabel Kotak Masuk	69
Gambar 4.9 Rancangan Antarmuka Tabel Upload Baju	70

1. Pendahuluan

1.1 Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) ini bertujuan untuk mendefinisikan merancang perangkat lunak yang akan dikembangkan. Dokumen tersebut akan digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap berikutnya.

1.2 Lingkup Masalah

Perangkat Lunak MEPAMS dikembangkan dengan tujuan untuk :

1. Memberikan pelayanan dan mengelola proses pemesanan produk.
2. Memberikan pelayanan *rate, comment, and share* produk.
3. Menangani pencarian data produk.
4. Melayani pengelolaan data pelanggan.
5. Memberikan pelayanan dan mengelola saran/keluhan.
6. Mengelola proses pemasaran produk.

Dan berjalan pada lingkungan dengan platform Android untuk aplikasi mobile dan untuk aplikasi web dapat berjalan dengan web browser.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-MEPAMS-XXX	Kode yang merepresentasikan kebutuhan pada MEPAMS (Pamella Swalayan) dimana XXX merupakan nomor fungsi produk.
MEPAMS	Perangkat lunak pengelolaan pada mobile dan web.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
Pamella Swalayan	Pamella Swalayan merupakan sebuah toko swalayan yang berada di Yogyakarta.

Member	Pelanggan dari Pamela Swalayan yang bertransaksi di swalayan melalui aplikasi
e-Commerce	Merupakan proses pembelian ataupun penjualan dari jasa maupun barang melalui jaringan internet atau <i>online</i>
Hybrid Mobile Application	Perangkat lunak mobile aplikasi yang dibangun menggunakan bahasa pemrograman HTML 5 dan Javascript.
CRM (<i>Customer Relationship Management</i>)	Merupakan strategi bisnis dalam mengelola hubungan korporasi dengan pelanggan pada level bisnis sehingga dapat memaksimalkan komunikasi dan meningkatkan pelayanan.
Framework of Dynamic CRM	Metode <i>Customer Relationship Management</i> untuk menentukan fitur dalam pembangunan perangkat lunak
SWOT	Metode analisis untuk menentukan strategi yang didasarkan pada kekuatan, kelemahan, peluang, dan ancaman
Market Basket Analysis	Metode yang dipakai dalam menganalisis keranjang belanja untuk menentukan keterkaitan barang yang dibeli pelanggan menggunakan <i>association rules</i> dan algoritma apriori

1.4 Referensi

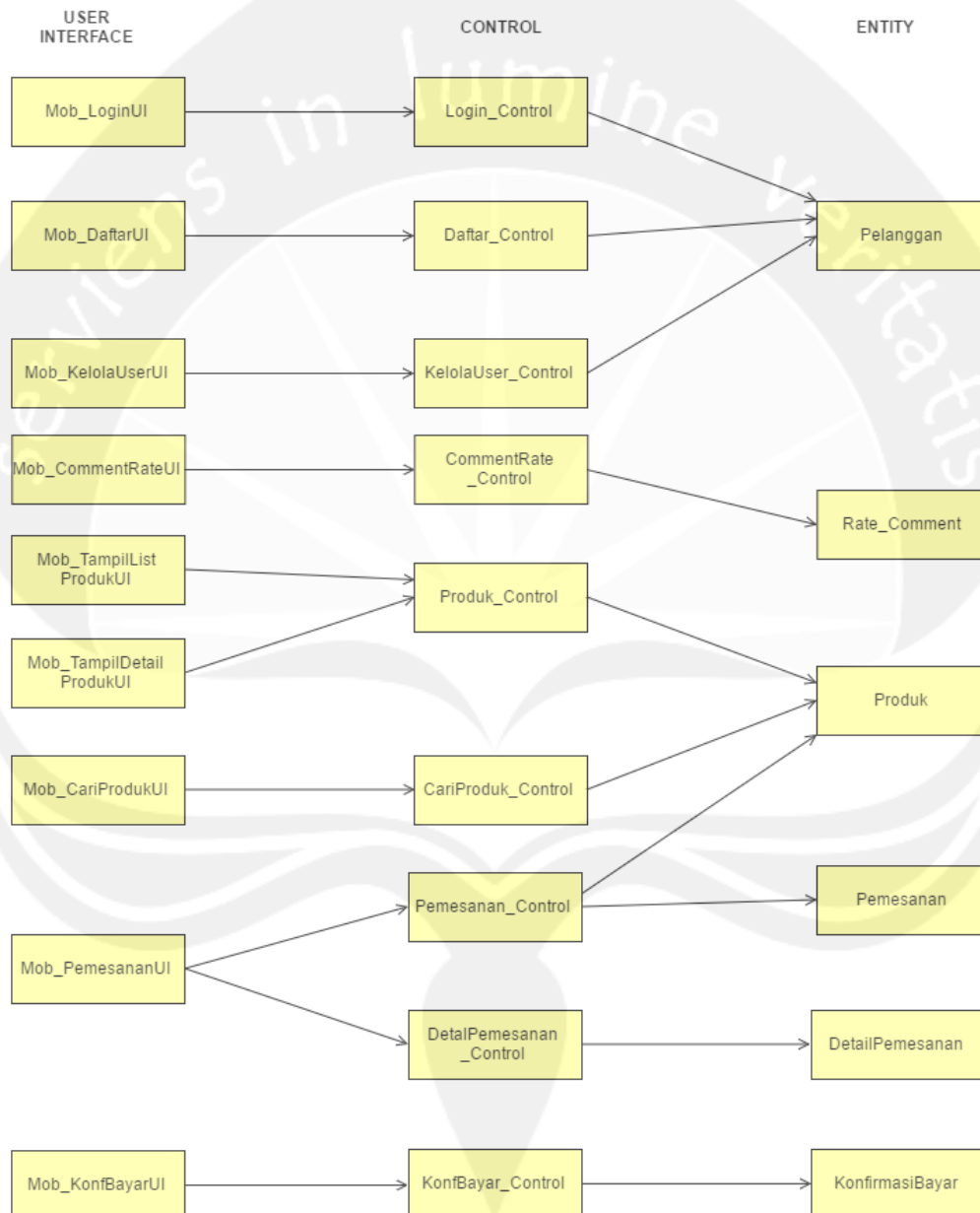
Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Yakobos Agung Yudha Pratama, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) MEPAMS*, Universitas Atma Jaya Yogyakarta.

2 Perancangan Sistem

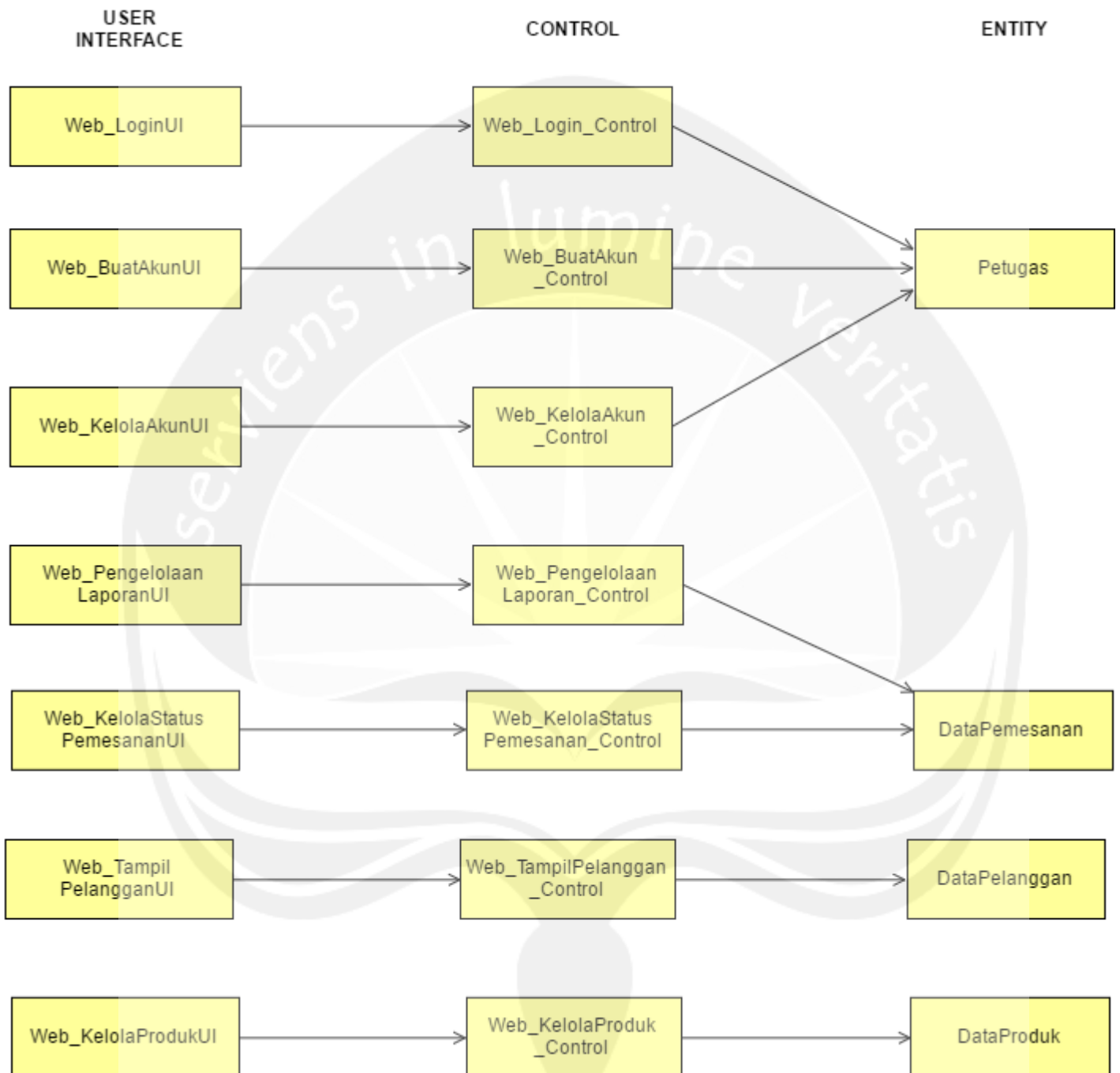
2.1 Perancangan Asitektur

2.1.1 Mobile Application



Gambar 2.1 Rancangan Arsitektur MEPAMS Mobile Application

2.1.2 Web Application

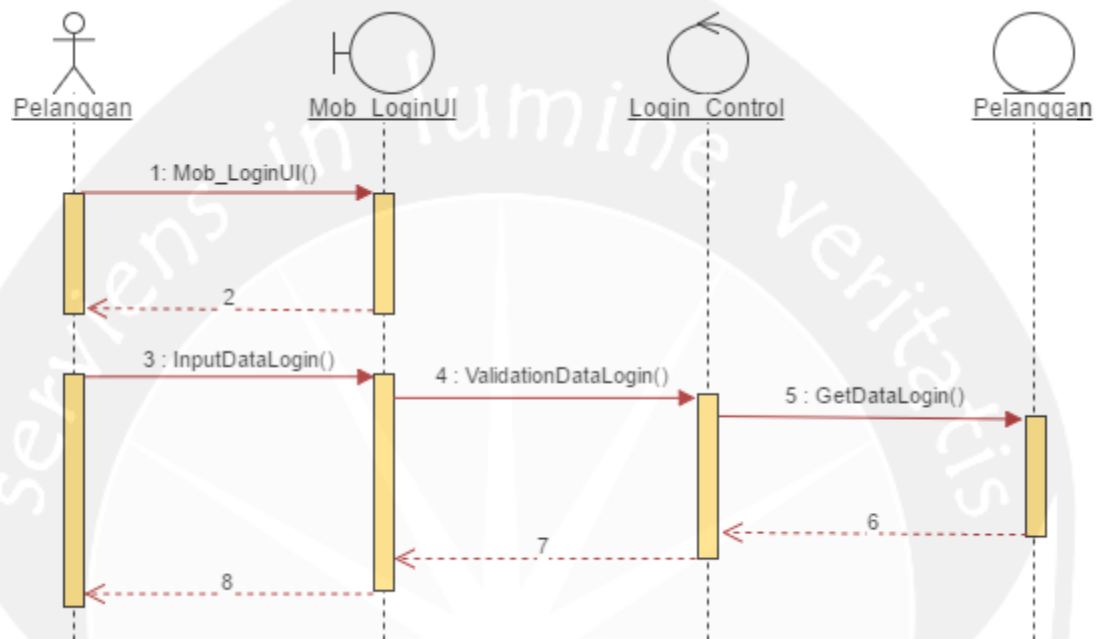


Gambar 2.2 Rancangan Arsitektur MEPAMS Web Application

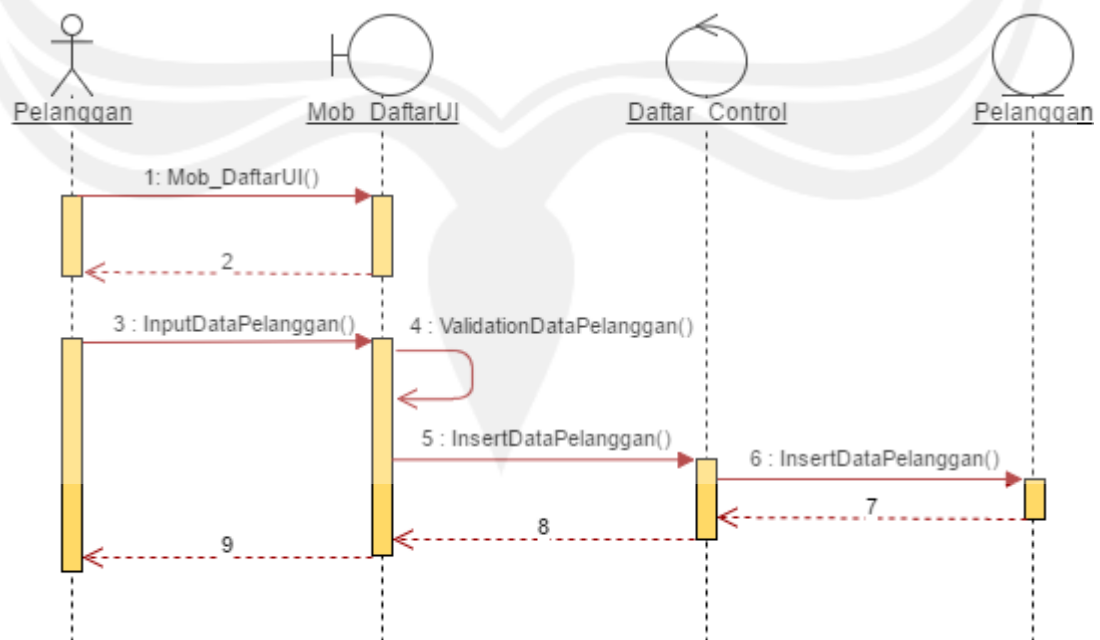
2.2 Perancangan Rinci

2.2.1. Sequence Diagram

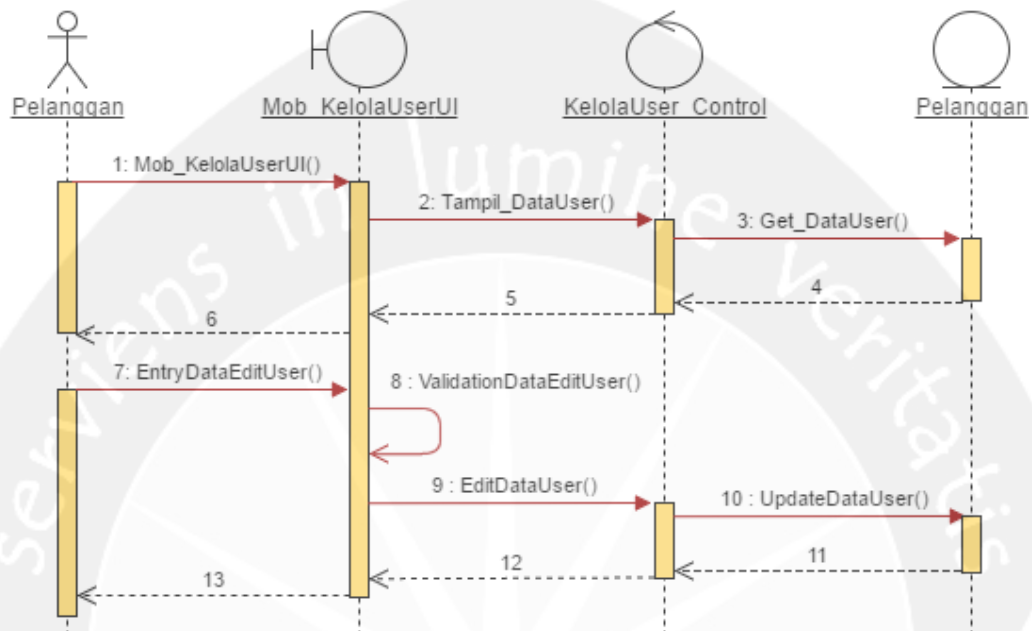
2.2.1.1 Login Mobile



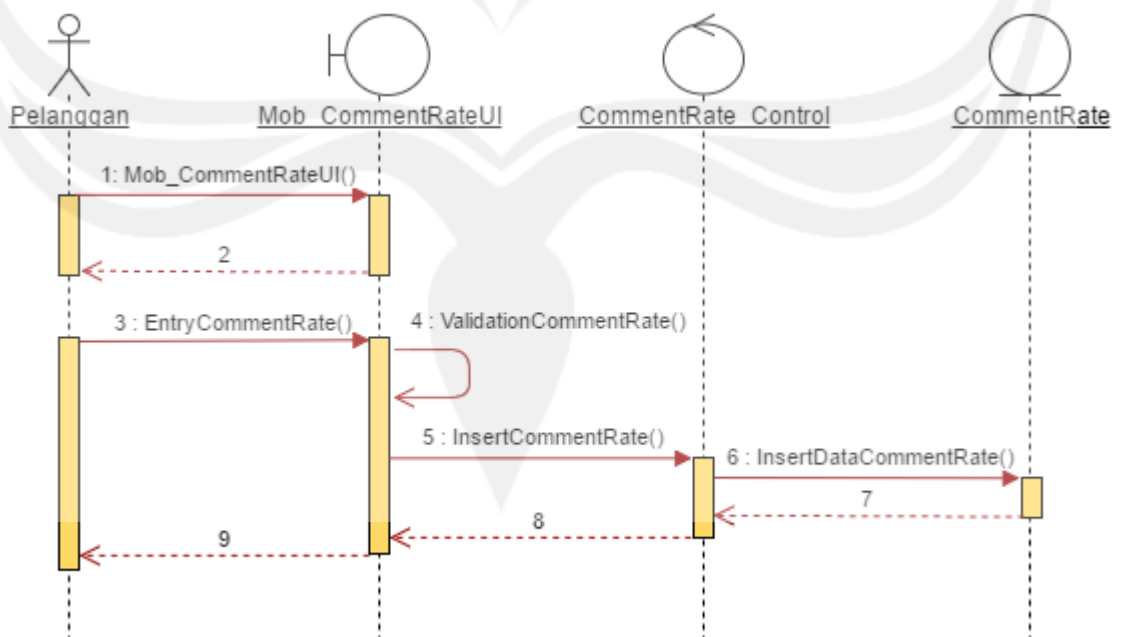
2.2.1.2 Registrasi / Mendaftar



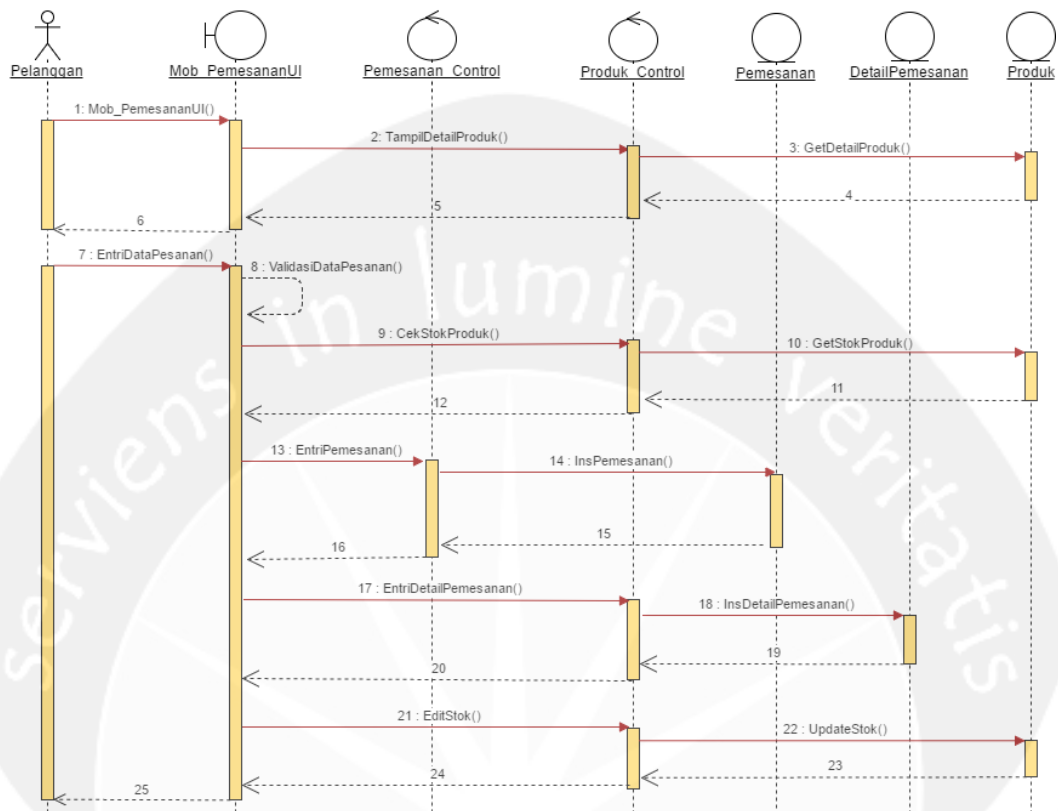
2.2.1.3 Pengelolaan Data Pelanggan



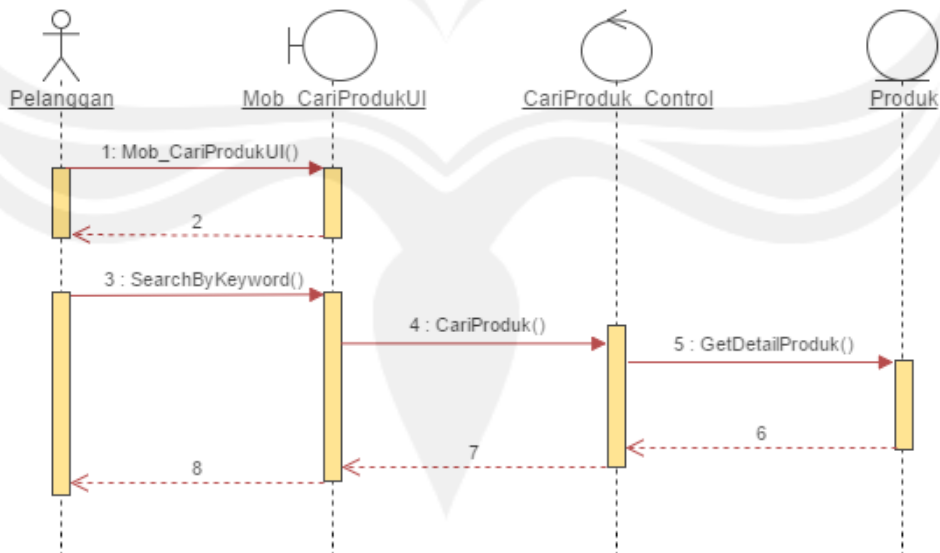
2.2.1.4 Fungsi Insert Rate & Share



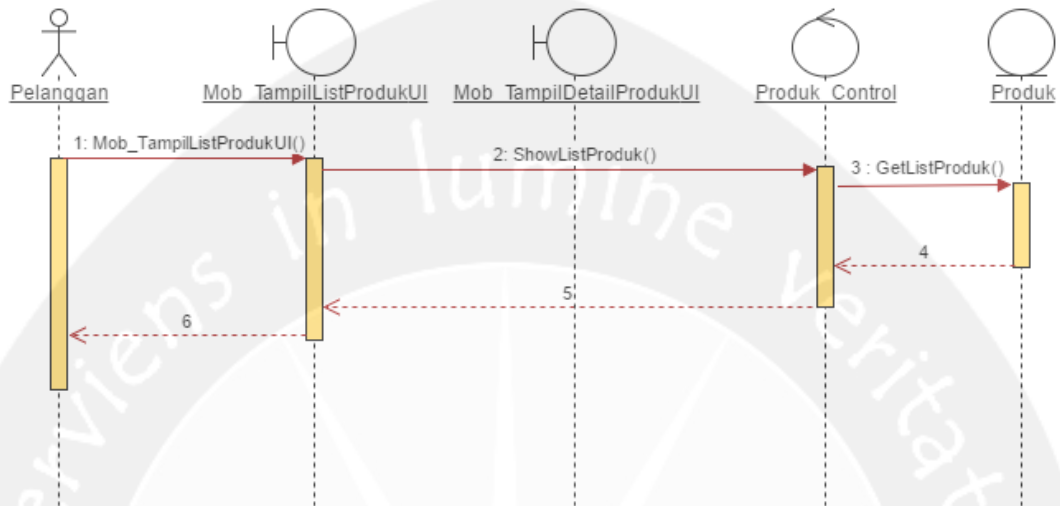
2.2.1.5 Fungsi Kelola Pemesanan



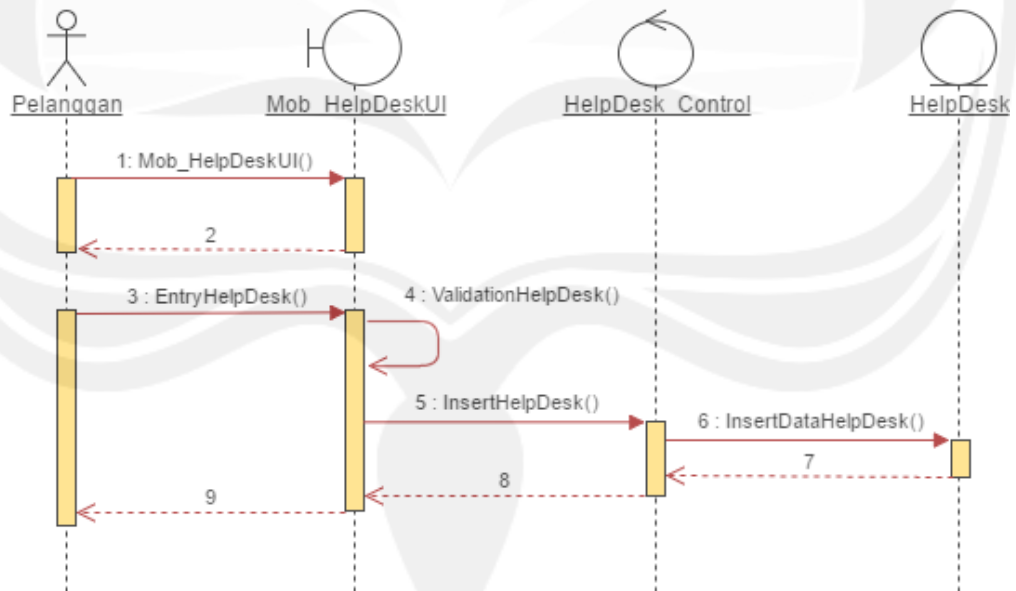
2.2.1.6 Fungsi Cari Produk



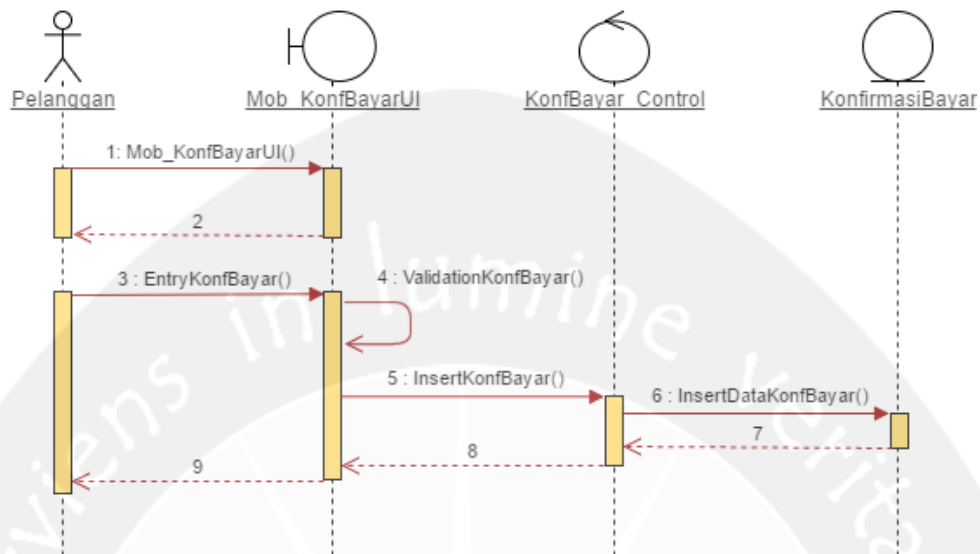
2.2.1.7 Fungsi Tampil Produk



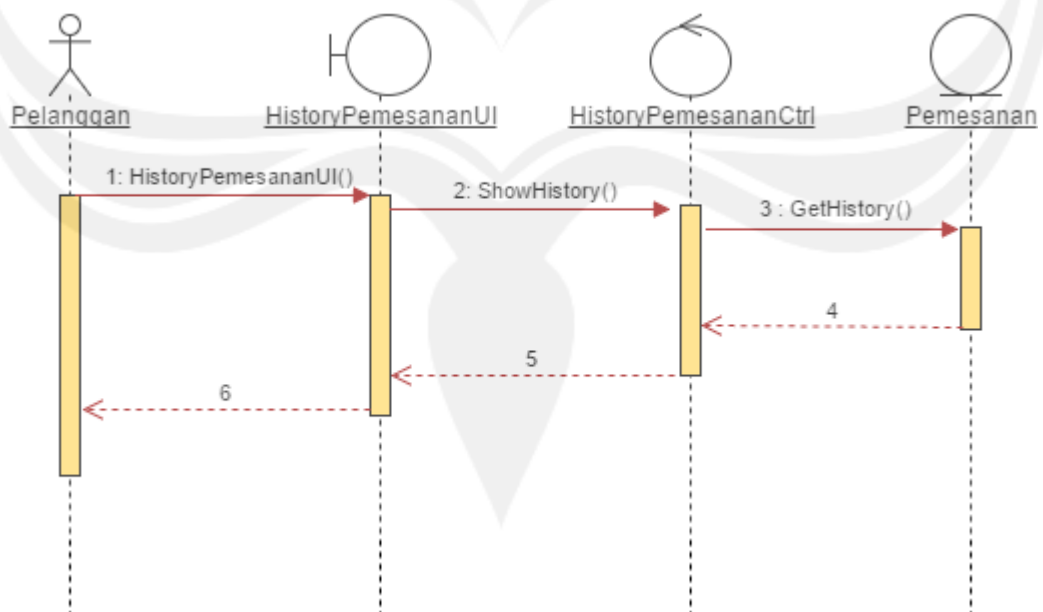
2.2.1.9 Fungsi Kelola Saran, Keluhan



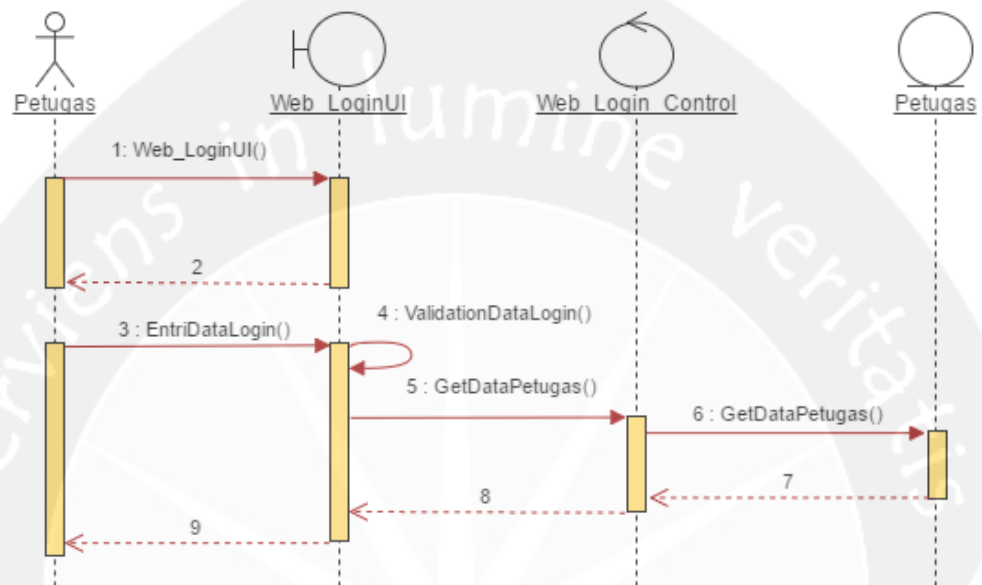
2.2.1.10 Fungsi Konfirmasi Pembayaran



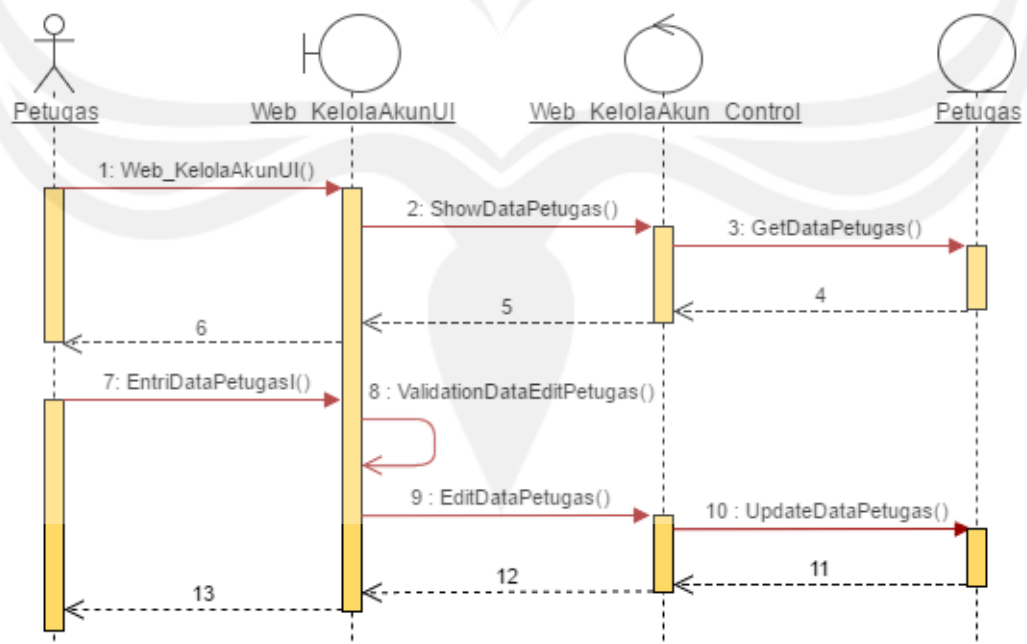
2.2.1.11 History Pemesanan



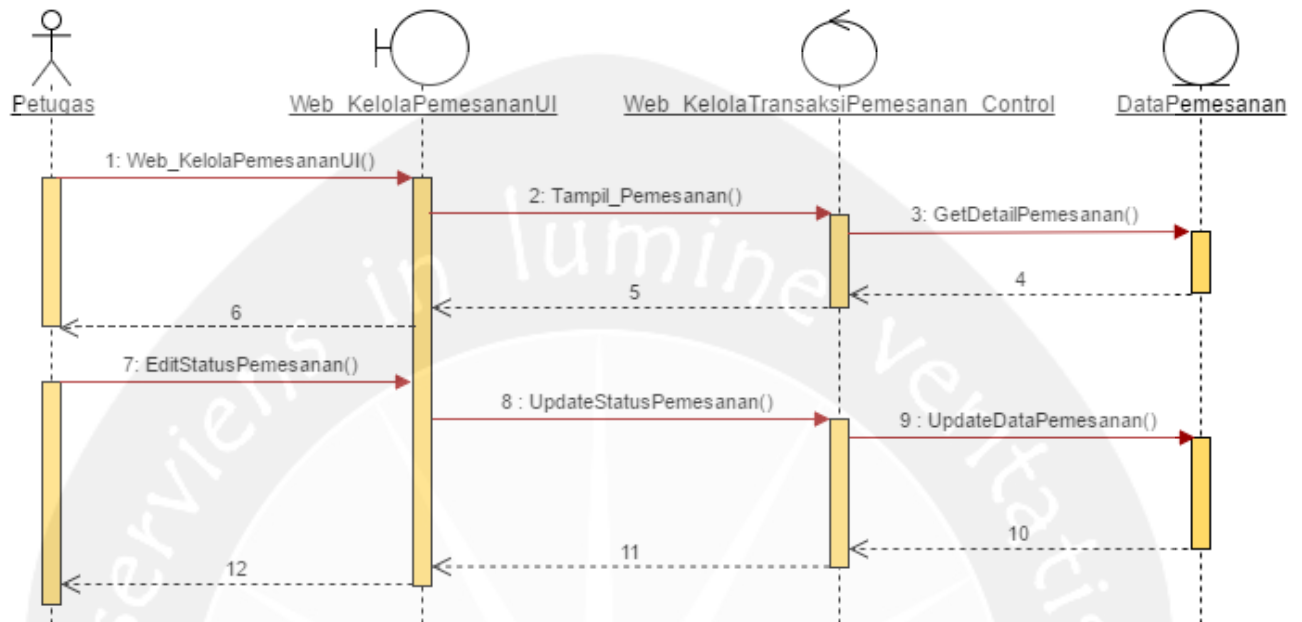
2.2.1.12 Fungsi Login Petugas



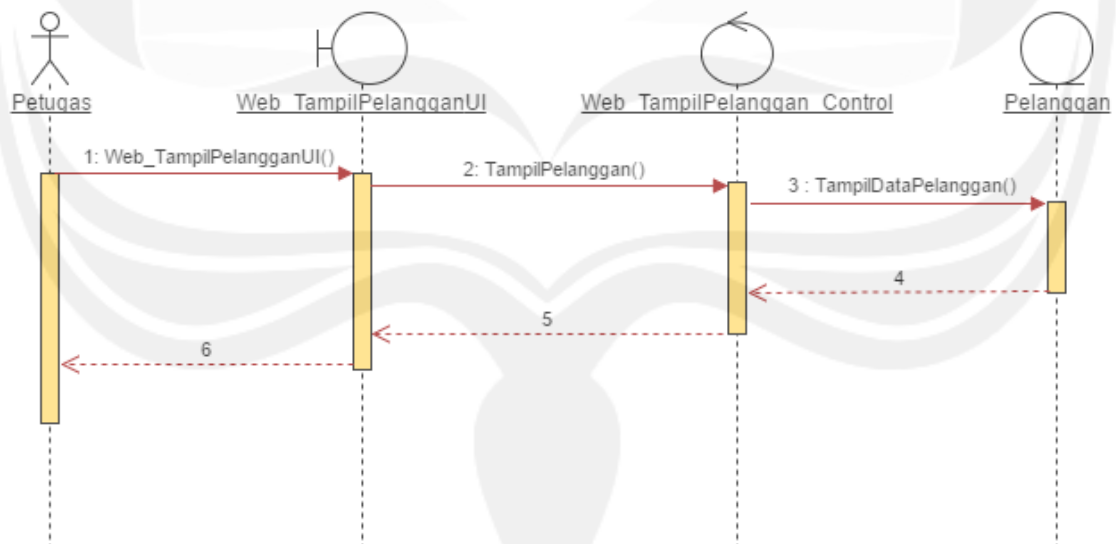
2.2.1.13 Fungsi Kelola Data Petugas



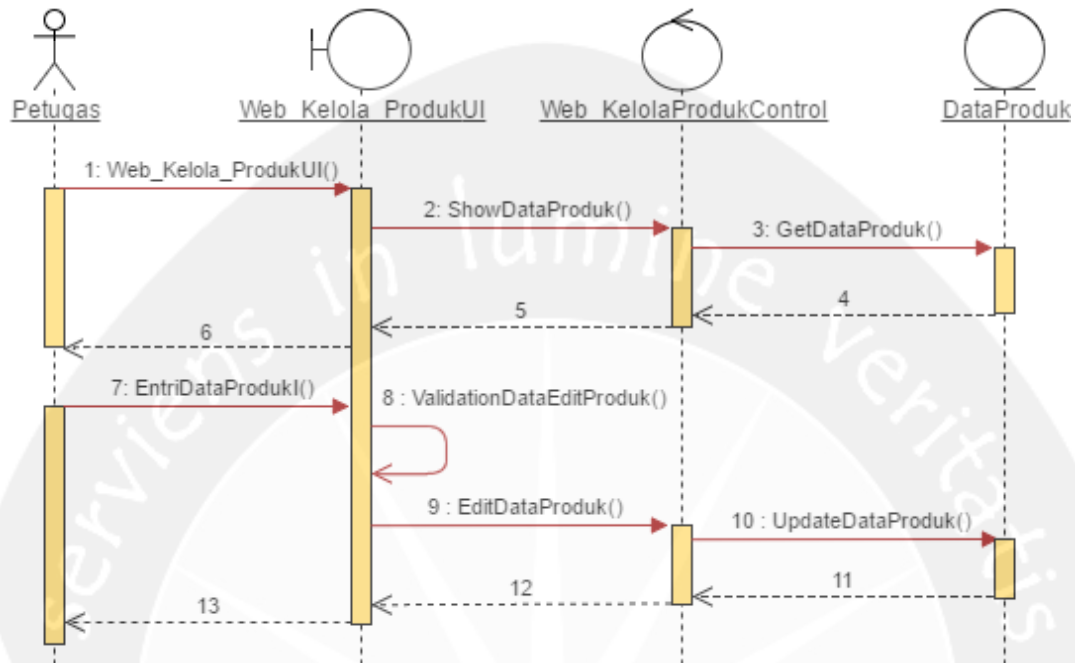
2.2.1.14 Fungsi Pengelolaan Transaksi



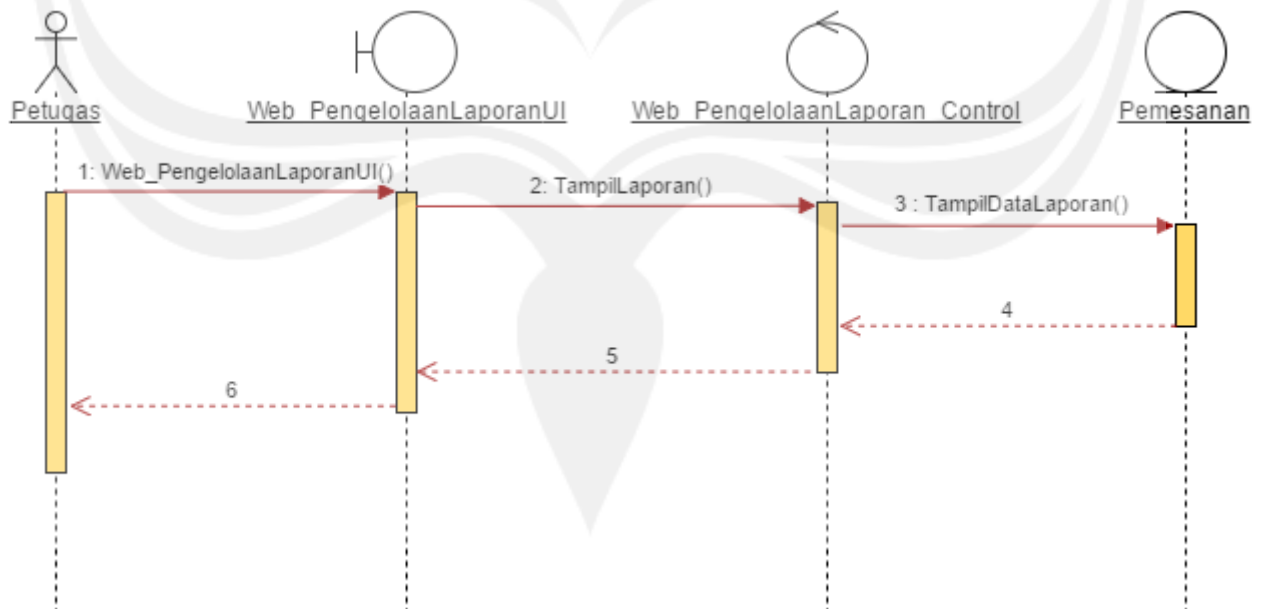
2.2.1.15 Fungsi Lihat Pelanggan



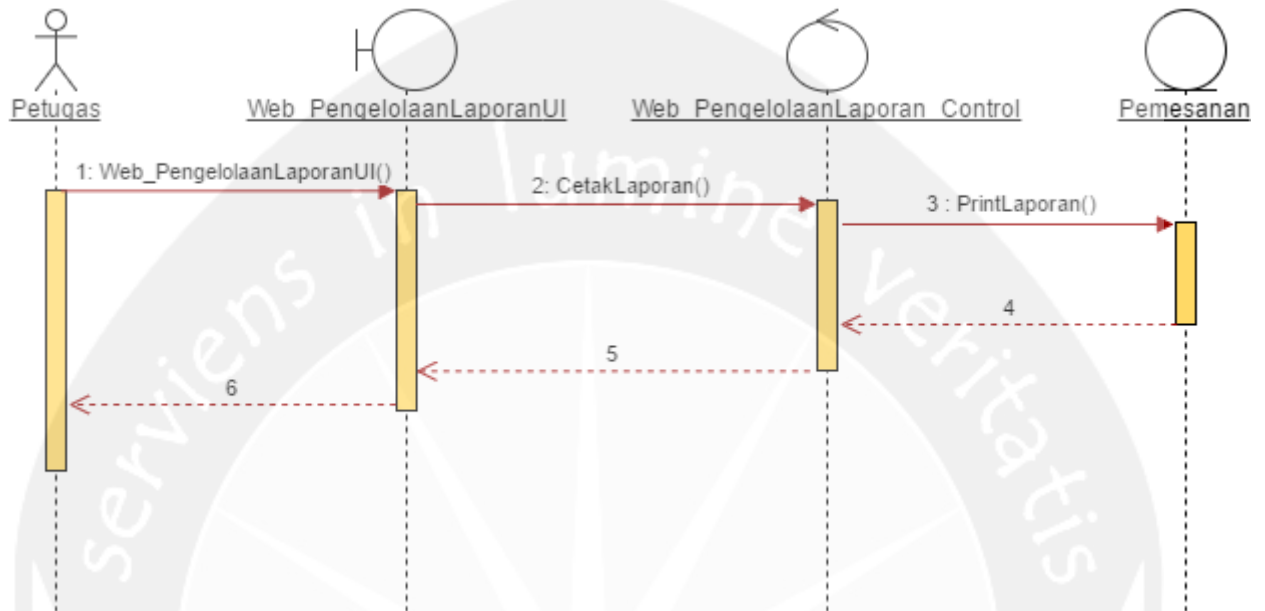
2.2.1.16 Fungsi Pengelolaan Produk



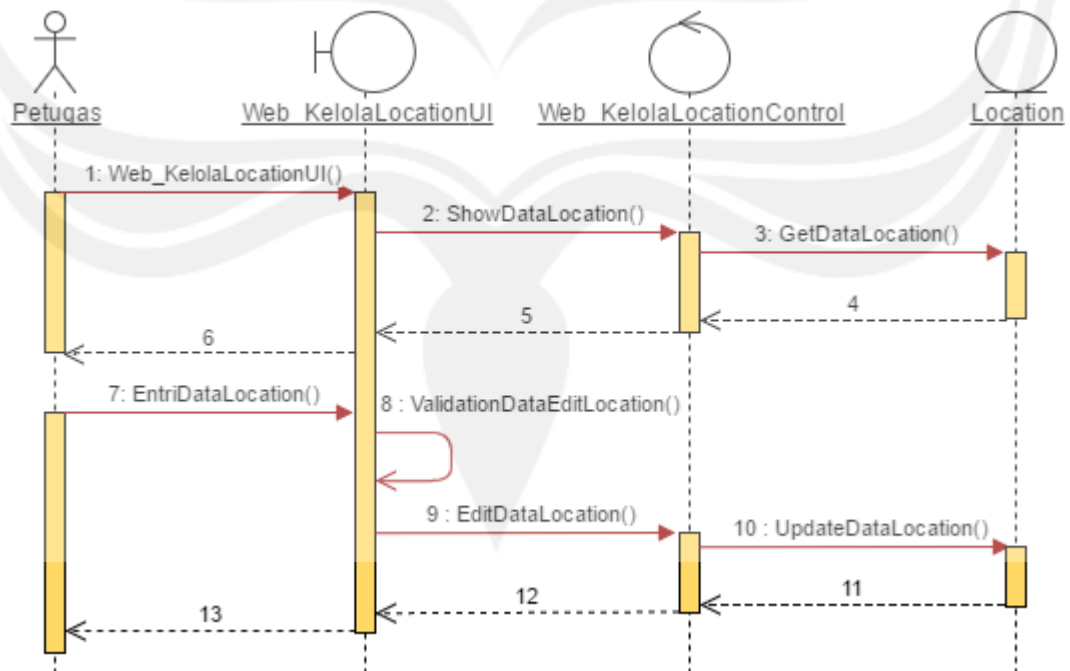
2.2.1.17 Fungsi Lihat Laporan



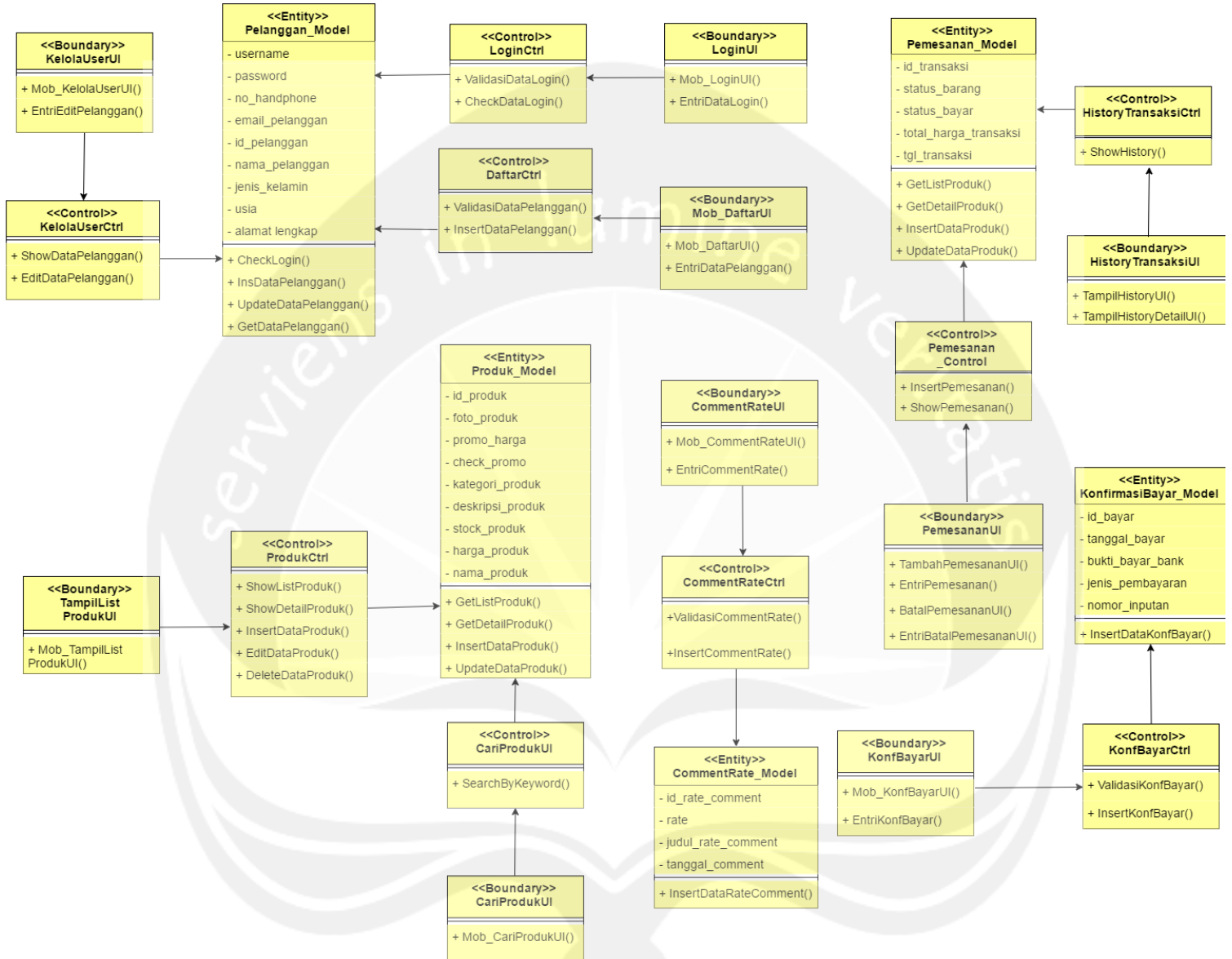
2.2.1.18 Fungsi Cetak Laporan



2.2.1.19 Fungsi Pengelolaan Location

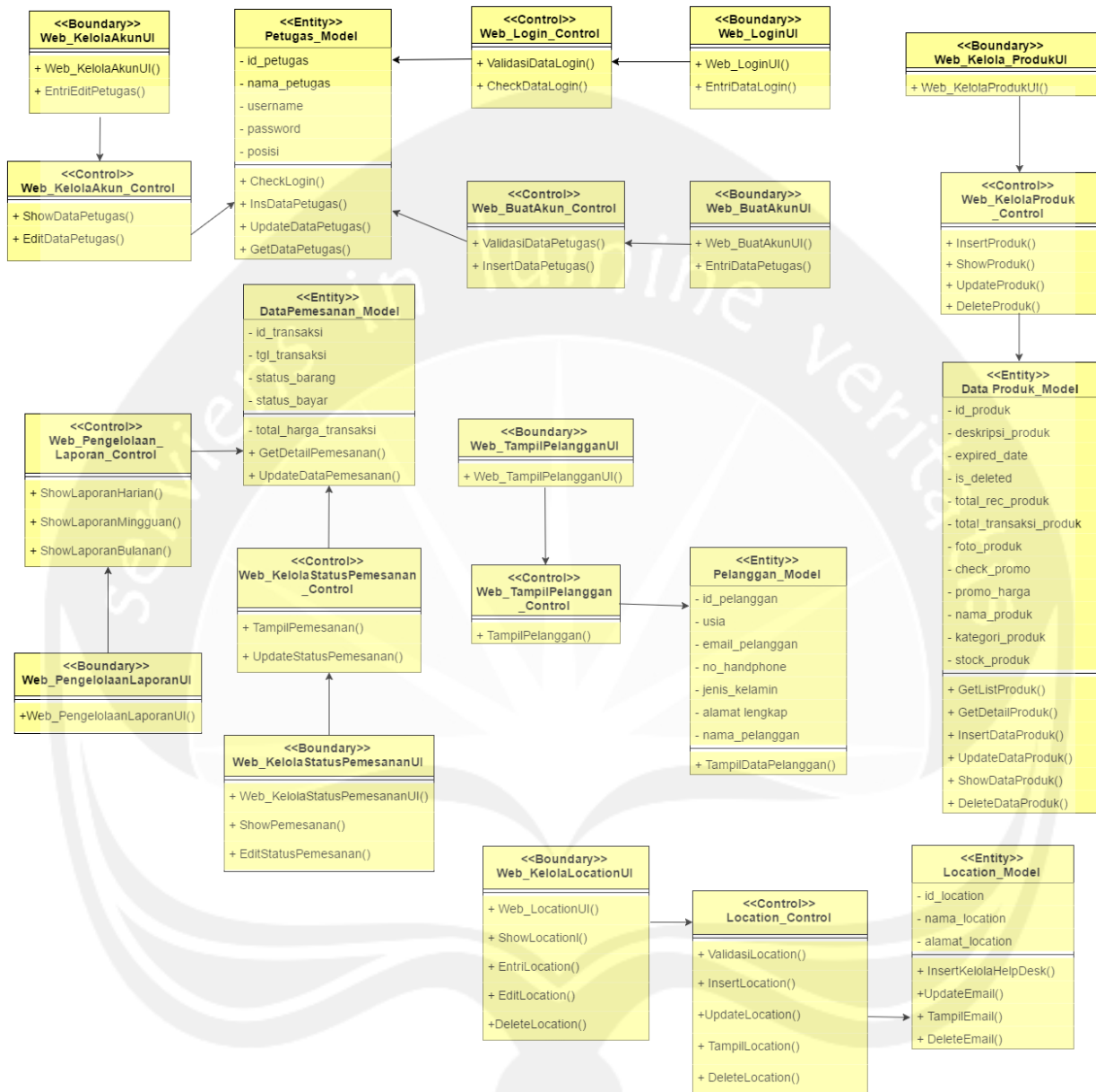


2.2.1.20 Class Diagram Mobile App



Gambar 2.24 Mobile Class Diagram

2.2.1.21 Class Diagram Web App



Gambar 2.25 Web Class Diagram

2.2.2 Class Diagram Specific Descriptions

Mobile Application

2.2.2.1 Specific Design Class DaftarUI

DaftarUI	<<boundary>>
+ Mob_DaftarUI() Prosedur ini digunakan untuk melakukan pemanggilan user interface daftar menjadi pelanggan + EntriDataPelanggan() Prosedur ini digunakan untuk melakukan input data calon pelanggan	

2.2.2.2 Specific Design Class KelolaUserUI

KelolaUserUI	<<boundary>>
+ Mob_KelolaUserUI() Prosedur ini digunakan untuk melakukan pemanggilan user interface kelola data pelanggan. + EntriEditPelanggan() Prosedur ini digunakan untuk edit data pelanggan.	

2.2.2.3 Specific Design Class CariProdukUI

CariProdukUI	<<boundary>>
+ Mob_CariProdukUI() Prosedur ini digunakan untuk melakukan pencarian berdasarkan kata kunci yang diisi oleh pelanggan	

2.2.2.4 Specific Design Class TampilListProdukUI

TampilListProdukUI	<<boundary>>
---------------------------	---------------------------------

+ Mob_TampilListProdukUI()
Prosedur ini digunakan untuk menampilkan interface list produk kepada pelanggan

2.2.2.5 Specific Design Class CommentRateUI

CommentRateUI	<<boundary>>
+ Mob_CommentRateUI() Prosedur ini digunakan untuk menampilkan interface comment and rate kepada pelanggan + EntriCommentRate() Prosedur ini digunakan untuk input comment rate dari pelanggan	

2.2.2.6 Specific Design Class KonfBayarUI

KonfBayarUI	<<boundary>>
+ Mob_KonfBayarUI() Prosedur ini digunakan untuk menampilkan interface konfirmasi pembayaran oleh pelanggan + EntriKonfBayar() Prosedur ini digunakan untuk melakukan entri data konfirmasi pembayaran oleh pelanggan	

2.2.2.7 Specific Design Class PemesananUI

PemesananUI	<<boundary>>
+ TambahPemesananUI() Prosedur ini digunakan untuk menampilkan kolom	

pemesanan produk oleh pelanggan

+ EntriPemesanan()

Prosedur ini digunakan untuk melakukan entri data pemesanan oleh pelanggan

+ BatalPemesananUI()

Prosedur ini digunakan untuk melakukan entri batal pemesanan oleh pelanggan

+ EntriBatalPemesananUI()

Prosedur ini digunakan menampilkan kolom batal pesan produk oleh pelanggan

2.2.2.9 Specific Design Class HistoryTransaksiUI

HistoryTransaksiUI	<<boundary>>
+ TampilHistoryUI() Prosedur yang digunakan untuk menampilkan history pemesanan	
+ TampilHistoryDetail() Prosedur yang digunakan untuk menampilkan detail dari tiap transaksi pemesanan	

2.2.2.10 Specific Design Class LoginUI

LoginUI	<<boundary>>
+ Mob_LoginUI() Prosedur ini digunakan untuk melakukan pemanggilan user interface login	
+ EntriDataLogin() Prosedur ini digunakan untuk melakukan input data login di level interface	

2.2.2.11 Specific Design Class LoginCtrl

LoginCtrl	<<control>>
+ ValidasiDataLogin() Prosedur yang digunakan untuk melakukan validasi input login user.	
+ CheckDataLogin() Prosedur yang digunakan untuk melakukan check terhadap data login user.	

2.2.2.12 Specific Design Class KelolaUserCtrl

KelolaUserCtrl	<<control>>
+ ShowDataPelanggan() Prosedur untuk menampilkan data pribadi pelanggan	
+ EditDataPelanggan() Prosedur ini digunakan untuk melakukan edit terhadap data pelanggan.	

2.2.2.13 Specific Design Class DaftarCtrl

DaftarCtrl	<<control>>
+ ValidasiDataPelanggan() Prosedur ini digunakan untuk melakukan pengecekan apakah semua inputan untuk mendaftar sudah di isi	
+ InsertDataPelanggan() Prosedur ini digunakan untuk melakukan insert data pelanggan ke dalam database	

2.2.2.14 Specific Design Class HistoryCtrl

HistoryTransaksiCtrl	<<control>>
+ShowHistory() Prosedur yang digunakan mengambil data pemesanan dari server.	

2.2.2.15 Specific Design Class PemesananCtrl

PemesananCtrl	<<control>>
+ InsertPemesanan() Prosedur yang digunakan untuk menyimpan data pemesanan ke database + ShowPemesanan() Prosedur yang digunakan untuk menampilkan data pemesanan sebelum di simpan di datase	

2.2.2.17 Specific Design Class ProdukCtrl

ProdukCtrl	<<control>>
+ ShowListProduk() Prosedur ini digunakan untuk menampilkan list produk + ShowDetailProduk() Prosedur ini digunakan untuk menampilkan detail produk + InsertDataProduk() Prosedur ini digunakan untuk menyimpan pesanan produk dari pelanggan + EditDataProduk() Prosedur ini digunakan untuk mengedit jumlah pesanan produk dari pelanggan	

+ DeleteDataProduk()

Prosedur ini digunakan untuk menghapus pesanan produk sebelum disimpan di database.

2.2.2.19 Specific Design Class KonfBayarCtrl

KonfBayarCtrl	<<control>>
+ ValidasiKonfBayar() Prosedur yang digunakan untuk mengecek data yang di input sudah benar atau tidak	
+ InsertKonfBayar Prosedur yang digunakan untuk menyimpan konfirmasi yang dilakukan oleh pelanggan	

2.2.2.20 Specific Design Class CommentRateCtrl

CommentRateCtrl	<<control>>
+ ValidasiCommentRate() Prosedur yang digunakan untuk mengecek data komentar yang di input sudah benar atau tidak	
+ InsertCommentRate Prosedur yang digunakan untuk menyimpan komentar yang dilakukan oleh pelanggan	

2.2.2.22 Specific Design Class Pelanggan_Model

Pelanggan_Model	<<entity>>
- Username: string, digunakan untuk menyimpan username pelanggan	
- Password: string, digunakan untuk menyimpan password pelanggan yang sudah di enkripsi	
- No_handphone: number, digunakan untuk menyimpan no handphone pelanggan	

- Email_pelanggan: string, digunakan untuk menyimpan alamat email pelanggan
- Id_pelanggan: int, digunakan untuk menyimpan id pelanggan
- Nama_pelanggan: string, digunakan untuk menyimpan nama pelanggan
- Jenis_kelamin: string, digunakan untuk menyimpan jenis_kelaim pelanggan
- Tanggal_lahir: date, digunakan untuk menyimpan tanggal lahir pelanggan
- Alamat_lengkap: string, digunakan untuk menyimpan alamat pelanggan

+ CheckLogin()

Prosedur ini digunakan untuk melakukan validasi login

+ InsDataPelanggan()

Prosedur ini digunakan untuk menyimpan data calon pelanggan ke dalam database

+ UpdateDataPelanggan()

Prosedur ini digunakan untuk mengupdate data pelanggan

+ GetDataPelanggan()

Prosdur ini digunakan untuk mengambil data pelanggan, untuk check login dan update data pelanggan.

2.2.2.23 Specific Design Class Pemesanan_Model

Pemesanan_Model	<<entity>>
<ul style="list-style-type: none"> - Id_transaksi: int, digunakan untuk menyimpan id transaksi - Status_barang: string, digunakan untuk menyimpan status barang pelanggan - Status_bayar: string, digunakan untuk menyimpan 	

<p>status pembayaran pelanggan.</p> <ul style="list-style-type: none"> - Total_harga_transaksi: decimal, digunakan untuk menyimpan total harga transaksi - Tanggal_transaksi: datetime, digunakan untuk menyimpan tanggal transaksi
<p>+ GetListProduk()</p> <p>Prosedur ini digunakan untuk mendapatkan list produk dari database</p> <p>+ GetDetailProduk()</p> <p>Prosedur ini digunakan untuk mendapatkan detail produk dari produk yang dipilih oleh pelanggan</p> <p>+ InsertDataProduk()</p> <p>Prosedur ini digunakan untuk menyimpan data produk pemesanan</p> <p>+ UpdateDataProduk()</p> <p>Prosedur ini digunakan untuk mengedit jumlah barang dari pesanan pelanggan sebelum disimpan di database</p>

2.2.2.24 Specific Design Class CommentRate_Model

CommentRate_Model	<<entity>>
<ul style="list-style-type: none"> - Id_rate_comment: int, digunakan untuk menyimpan id commentrate - Rate: int, digunakan untuk menyimpan jumlah rate produk - Judul_Rate_Comment: string, digunakan untuk menyimpan judul comment - Tanggal_Comment: datestring, digunakan untuk menyimpan tanggal comment 	
<p>+ InsertDataRateComment()</p> <p>Prosedur ini digunakan untuk menyimpan data comment dan rate dari produk yang dipilih pelanggan.</p>	

2.2.2.25 Specific Design Class KonfirmasiBayar_Model

KonfirmasiBayar_Model	<<entity>>
<ul style="list-style-type: none"> - Id_bayar: int, digunakan untuk menyimpan id pembayaran - Tanggal_bayar: datetime, digunakan untuk menyimpan tanggal pembayaran - Nominal_pembayaran: decimal, digunakan untuk menyimpan nominal pembayaran 	
+ InsertDataKonfBayar() Prosedur yang digunakan untuk menyimpan data konfirmasi pembayaran ke dalam database	

2.2.2.27 Specific Design Class Produk_Model

Produk_Model	<<entity>>
<ul style="list-style-type: none"> - Id_produk: int, digunakan untuk menyimpan id produk - Foto_produk: string, digunakan untuk menyimpan alamat pengaksesan foto produk - diskon_promo: float, digunakan untuk menyimpan diskon promo - Check_promo: int, digunakan untuk menyimpan check promo - Kategori_produk: string, digunakan untuk menyimpan sub kategori dari produk - Deskripsi_produk: string, digunakan untuk untuk menyimpan deskripsi produk - Stock_produk: int, digunakan untuk menyimpan stock produk - Harga_produk: decimal, digunakan untuk menyimpan 	

<p>harga produk</p> <p>- Nama_produk: string, digunakan untuk menyimpan nama produk</p>
<p>+ GetListProduk()</p> <p>Prosedur ini digunakan untuk mendapatkan list produk dari database produk</p> <p>+ GetDetailProduk()</p> <p>Prosedur ini digunakan untuk mendapatkan detail produk</p>

Web Application

2.2.2.29 Specific Design Class Web_LoginUI

Web_LoginUI	<<boundary>>
<p>+ Web_LoginUI()</p> <p>Prosedur ini digunakan untuk menampilkan halaman login</p> <p>+ EntriDataLogin()</p> <p>Prosedur ini digunakan untuk mengambil data yang di isi user untuk di validasi login</p>	

2.2.2.30 Specific Design Class Web_KelolaAkunUI

Web_KelolaAkunUI	<<boundary>>
<p>+ ShowDataPetugas()</p>	

Prosedur ini digunakan untuk menampilkan data petugas
+ EntriDataPetugas()
Prosedur ini digunakan untuk menginput data petugas

2.2.2.31 Specific Design Class Web_KelolaProdukUI

Web_KelolaProdukUI	<<boundary>>
+ Web_KelolaProdukUI() Prosedur ini digunakan untuk menampilkan pengelolaan produk	

2.2.2.32 Specific Design Class Web_BuatAkunUI

Web_BuatAkunUI	<<boundary>>
+ Web_BuatAkunUI() Prosedur ini digunakan untuk menampilkan view pembuatan akun petugas + EntriDataPetugas() Prosedur ini digunakan untuk menginputkan data petugas baru	

2.2.2.33 Specific Design Class Web_TampilPelangganUI

Web_TampilPelangganUI	<<boundary>>
+ Web_TampilPelangganUI() Prosedur ini digunakan untuk menampilkan data pelanggan	

2.2.2.34 Specific Design Class Web_PengelolaanLaporanUI

Web_PengelolaanLaporanUI	<<boundary>>
+ Web_PengelolaanLaporanUI() Prosedur ini digunakan untuk menampilkan pengelolaan laporan	

2.2.2.35 Specific Design Class Web_KelolaHelpDeskUI

Web_KelolaHelpDeskUI	<<boundary>>
+ Web_KelolaHelpDeskUI() Prosedur ini digunakan untuk menampilkan halaman helpdesk + TampilKelolaHelpDesk() Prosedur ini digunakan untuk menampilkan pengelolaan helpdesk dalam table + UpdateKelolaHelpDesk() Prosedur ini digunakan untuk mengedit pengelolaan helpDesk	

2.2.2.36 Specific Design Class Web_KelolaPemesananUI

Web_KelolaPemesananUI	<<boundary>>
+ Web_KelolaStatusPemesananUI() Prosedur ini digunakan untuk menampilkan halaman pengelolaan transaksi + ShowPemesanan() Prosedur ini digunakan untuk menampilkan pemesanan ke dalam table	

2.2.2.37 Specific Design Class Web_KelolaLocationUI

Web_KelolaLocationUI	<<boundary>>
+ Web_LocationUI() Prosedur ini digunakan untuk menampilkan location UI + ShowLocation() Prosedur ini digunakan untuk menampilkan location di dalam table	

2.2.2.38 Specific Design Class Web_EmailUI

Web_KelolaLocationUI	<<boundary>>
+ Web_EmailUI() Prosedur ini digunakan untuk menampilkan halaman emailUI	

2.2.2.39 Specific Design Class Web_Login_Control

Web_Login_Control	<<control>>
+ ValidasiDataLogin() Prosedur ini digunakan untuk memvalidasi data login	

2.2.2.40 Specific Design Class Web_KelolaAkun_Control

Web_KelolaAkun_Control	<<control>>
+ ShowDataPetugas() Prosedur ini digunakan untuk menampilkan data petugas + EditDataPetugas()	

Prosedur ini digunakan untuk mengedit data petugas

2.2.2.41 Specific Design Class Web_BuatAkun_Control

Web_BuatAkun_Control	<<control>>
+ ValidasiDataPetugas() Prosedur ini digunakan untuk memvalidasi data petugas apakah sudah diisi dengan benar + InsertDataPetugas() Prosedur ini digunakan untuk menginsert data petugas	

2.2.2.42 Specific Design Class Web_KelolaProduk_Control

Web_KelolaProduk_Control	<<control>>
+ InsertProduk() Prosedur ini digunakan untuk menginput produk baru + UpdateProduk() Prosedur ini digunakan untuk mengupdate produk + DeleteProduk() Prosedur ini digunakan untuk mengnonaktifkan produk + TampilProduk() Prosedur ini digunakan untuk menampilkan produk + UpdateProduk() + DeleteProduk()	

2.2.2.43 Specific Design Class

Web_KelolaPemesanan_Control

Web_PengelolaanLaporan_Control	<<control>>

+ Web_KelolaPemesanan()

Prosedur ini digunakan untuk menampilkan pemesanan

2.2.2.44 Specific Design Class Web_Location_Control

Web_TampilPelanggan_Control	<<control>>
+ InsertLocation() Prosedur ini digunakan untuk menginsert location + UpdateLocation() Prosedur ini digunakan untuk mengupdate location + TampilLocation() Prosedur ini digunakan untuk menampilkan location + DeleteLocation() Prosedur ini digunakan untuk menonaktifkan location	

2.2.2.45 Specific Design Class Web_Email_Control

Web_Email_Control	<<control>>
+ InsertEmail() Prosedur ini digunakan untuk menginsert email + UpdateEmail() Prosedur ini digunakan untuk mengedit email	

2.2.2.46 Specific Design Class Web_KelolaHelpDeskUI

Web_KelolaHelpDeskUI	<<control>>
Prosedur ini digunakan untuk mengupdate helpdesk	

2.2.2.47 Specific Design Class

Web_TampilPelanggan_Control

Web_TampilPelangganControl	<<control>>
+ TampilPelanggan() Prosedur ini digunakan untuk menampilkan data pelanggan	

2.2.2.48 Specific Design Class

Web_Pengelolaan_Laporan_Control

Web_Pengelolaan_Laporan_Control	<<control>>
+ ShowLaporanHarian() Prosedur ini digunakan untuk menampilkan laporan harian + ShowLaporanMingguan() Prosedur ini digunakan untuk menampilkan laporan mingguan + ShowLaporanBulanan() Prosedur ini digunakan untuk menampilkan laporan bulanan	

2.2.2.49 Specific Design Class Petugas_Model

Petugas_Model	<<entity>>
- Id_petugas: int, digunakan untuk menyimpan id petugas - Nama_petugas: string, digunakan untuk menyimpan nama petugas - Username: string, digunakan untuk menyimpan username petugas - Password: string, digunakan untuk menyimpan password - Posisi: string, digunakan untuk menyimpan posisi	

petugas
+ CheckLogin() Prosedur ini digunakan untuk melakukan pengecekan login + InsDataPetugas() Prosedur ini digunakan untuk menginsert data petugas + UpdateDataPetugas() Prosedur ini digunakan untuk mengupdate data petugas + GetDataPetugas() Prosedur ini digunakan untuk mendapatkan data petugas

2.2.2.50 Specific Design Class Data_Produk_Model

Data_Produk_Model	<<entity>>
<ul style="list-style-type: none"> - Id_produk: int, digunakan untuk menyimpan id produk - Deskripsi_produk: string, digunakan untuk menyimpan deskripsi produk - Is_deleted: int, digunakan untuk menyimpan status delete - Foto_produk: string, digunakan untuk menyimpan alamat pengaksesan foto - Check_promo: int, digunakan untuk menyimpan status promo - Diskon_promo: int, digunakan untuk menyimpan diskon promo - Nama_produk: string, digunakan untuk menyimpan nama produk - Kategori_produk: string, digunakan untuk menyimpan sub kategori produk - Stock_produk: int, digunakan untuk menyimpan stock produk - Stock_produk_sisa: int, digunakan untuk menyimpan 	

<p>stock produk sisa</p> <p>- Stock_produk_terjual: int, digunakan untuk menyimpan stock produk terjual</p>
<p>+ GetListProduk()</p> <p>Prosedur ini digunakan untuk menampilkan list produk</p> <p>+ InsertDataProduk()</p> <p>Prosedur ini digunakan untuk menginsert data produk</p> <p>+ UpdateDataProduk()</p> <p>Prosedur ini digunakan untuk mengupdate data produk</p> <p>+ ShowDataProduk()</p> <p>Prosedur ini digunakan untuk menampilkan data produk</p> <p>+ DeleteDataProduk()</p> <p>Prosedur ini digunakan untuk mengonaktifkan produk</p>

2.2.2.51 Specific Design Class DataPemesanan_Model

DataPemesanan_Model	<<entity>>
<p>- Id_transaksi: int, digunakan untuk menyimpan id transaksi</p> <p>- Tgl_transaksi: datetime, digunakan untuk menyimpan tanggal transaksi</p> <p>- Status_barang: string, digunakan untuk menyimpan status barang</p> <p>- Status_bayar: string, digunakan untuk menyimpan status pembayaran</p> <p>- Nominal_bayar: decimal, digunakan untuk menyimpan nominal pembayaran</p>	
<p>+ getDetailPemesanan()</p> <p>Prosedur ini digunakan untuk menampilkan detail pemesanan</p> <p>+ UpdateDataPemesanan()</p>	

Prosedur ini digunakan untuk mengedit data pemesanan

2.2.2.52 Specific Design Class Pelanggan_Model

Pelanggan_Model	<<entity>>
<ul style="list-style-type: none"> - Id_pelanggan: int, digunakan untuk menyimpan data id pelanggan - Tanggal_lahir: date, digunakan untuk menyimpan tanggal lahir - Email_pelanggan: string, digunakan untuk menyimpan email pelanggan - No_handphone: number, digunakan untuk menyimpan no handphone - Alamat_lengkap: string, digunakan untuk menyimpan alamat pelanggan - Nama_pelanggan: string, digunakan untuk menyimpan nama pelanggan 	
+ tampilkandatapelanggan()	
Prosedur ini digunakan untuk menampilkan data pelanggan	

2.2.2.53 Specific Design Class KelolaHelpDesk_Model

KelolaHelpDesk_Model	<<entity>>
<ul style="list-style-type: none"> - Id_kelola_helpdesk: int, digunakan untuk menyimpan id kelola helpdesk - Respon: string, digunakan untuk menyimpan respon helpdesk - Tanggal_respon: datetime, digunakan untuk menyimpan tanggal respon 	
+ UpdateKelolaHelpDesk()	
Prosedur ini digunakan untuk mengedit helpdesk	

2.2.2.54 Specific Design Class Email_Model

Email_Model	<<entity>>
<ul style="list-style-type: none"> - Id_email: int, digunakan untuk menyimpan id email - Tanggal_email: datetime, digunakan untuk menyimpan tanggal email - Isi_email: string, digunakan untuk menyimpan isi email - Judul_email: string, digunakan untuk menyimpan judul email 	
+ UpdateEmail() Prosedur ini digunakan untuk mengedir email + InsertEmail() Prosedur ini digunakan untuk menginsert email + TampilEmail() Prosedur ini digunakan untuk menampilkan email + DeleteEmail() Prosedur ini digunakan untuk menghapus email	

2.2.2.55 Specific Design Class Location_Model

Location_Model	<<entity>>
<ul style="list-style-type: none"> - Id_location: int, digunakan untuk menyimpan id lokasi - Longitude: float, digunakan untuk menyimpan longitude - Latitude: float, digunakan untuk menyimpan latitude - Nama_location: string, digunakan untuk menyimpan nama lokasi - Alamat_location: string, digunakan untuk menyimpan alamat lokasi 	
+ UpdateLocation() Prosedur ini digunakan untuk mengupdate lokasi + InsertLocation()	

Prosedur ini digunakan untuk menyimpan data lokasi

+ TampilLocation()

Prosedur ini digunakan untuk menampilkan lokasi

+ DeleteLocation()

Prosedur ini digunakan untuk menghapus lokasi



3. PERANCANGAN DATA

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas TBL_DETAIL_TRANSAKSI

Nama	Tipe	Panjang	Keterangan
id_detail_transaksi	Integer	11	Id detail transaksi
id_transaksi	Integer	11	Id transaksi pemesanan
Id_produk	Integer	11	Id produk barang
harga_barang_transaksi	Decimal	10.0	Harga barang produk
kuantitas	Integer	11	Kuantitas produk
subtotal	Decimal	10.0	Total harga dari kuantitas produk
rec_user_check	Integer	1	Check rekomendasi produk

3.1.2 Deskripsi Entitas TBL_KONFIRMASI_BAYAR

Nama	Tipe	Panjang	Keterangan
id_bayar	Integer	11	Id konfirmasi dari pembayaran
id_transaksi	Integer	11	Id transaksi dari pembayaran
id_pelanggan	Integer	11	Id pelanggan dari pembayaran
id_rekening_pamella	Integer	11	Id rekening pamella
nominal_bayar	Decimal	10.0	Nominal pembayaran
tgl_transfer	Datetime	-	Tanggal pelanggan melakukan transfer
tgl_konf_bayar	Datetime	-	Tanggal pelanggan melakukan konfirmasi pembayaran

3.1.3 Deskripsi Entitas TBL_PRODUK

Nama	Tipe	Panjang	Keterangan
id_produk	Integer	11	Id dari produk
nama_produk	Variabel Character	100	Nama dari produk
harga_produk	Decimal	10.0	Harga dari produk
stock_produk	Integer	11	Stock dari produk
stock_produk_sisa	Integer	11	Stock sisa dari produk
stock_produk_terjual	Integer	11	Stock terjual dari produk
deskripsi_produk	Variabel Character	200	Deskripsi dari produk
id_kategori	Integer	11	Id kategori dari produk
jenis_produk	Variabel Character	100	Jenis / sub kategori dari produk
check_promo	Integer	1	Check apakah produk ini promo
diskon_promo	Integer	11	Diskon jika produk promo
foto_produk	Variabel Character	255	Alamat untuk mengakses foto produk
check_delete	Integer	2	Check apakah produk terhapus atau tidak

3.1.4 Deskripsi Entitas TBL_TRANSAKSI

Nama	Tipe	Penjang	Keterangan
id_transaksi	Integer	11	Id dari transaksi
id_pelanggan	Integer	11	Id dari pelanggan
id_lokasi	Integer	11	Id dari lokasi pamella
tgl_transaksi	Datetime	-	Tanggal transaksi dilakukan
total_harga_transaksi	Decimal	10.0	Total harga barang transaksi

status_bayar	Variable Character	25	Status pembayaran dari transaksi
status_barang	Variable Character	25	Status barang dari transaksi

3.1.5 Deskripsi Entitas TBL_KATEGORI

Nama	Tipe	Panjang	Keterangan
id_kategori	Integer	11	Id dari kategori
nama_kategori	Variabel Character	50	Nama dari kategori

3.1.6 Deskripsi Entitas TBL_LOKASI

Nama	Tipe	Panjang	Keterangan
id_lokasi	Integer	11	Id dari lokasi
nama_lokasi	Variabel Character	50	Nama dari lokasi
longitude	Float	10.6	Longitude dari lokasi
latitude	Float	10.6	Latitude dari lokasi
alamat_lokasi	Variabel Character	100	Alamat dari lokasi

3.1.7 Deskripsi Entitas TBL_PELANGGAN

Nama	Type	Penjang	Keterangan
id_pelanggan	Integer	11	Id dari pelanggan
nama_pelanggan	Variabel Character	100	Nama dari pelanggan
jenis_kelamin	Variabel Character	25	jenis kelamin dari pelanggan
tanggal_lahir	Date	-	Tanggal lahir pelanggan
alamat_lengkap	Variabel Character	100	Alamat lengkap dari pelanggan
email_pelanggan	Variabel Character	100	Email dari pelanggan
no_handphone	Variabel Character	15	No handphone pelanggan
username	Variabel Character	50	Username milik pelanggan
password	Variabel Character	100	Password milik pelanggan
verifikasi_code	Integer	4	Verifikasi code pelanggan

3.1.8 Deskripsi Entitas TBL_PETUGAS

Nama	Type	Panjang	Keterangan
id_petugas	Integer	11	Id milik petugas
nama_petugas	Variabel Character	100	Nama milik petugas
posisi	Variabel Character	50	Posisi yang dimiliki oleh petugas
id_role	Integer	11	Id role dari petugas
username	Variabel Character	50	Username dari petugas

password	Variabel Character	100	Password dari petugas
check_delete	Integer	2	Check delete apakah petugas dihapus atau tidak

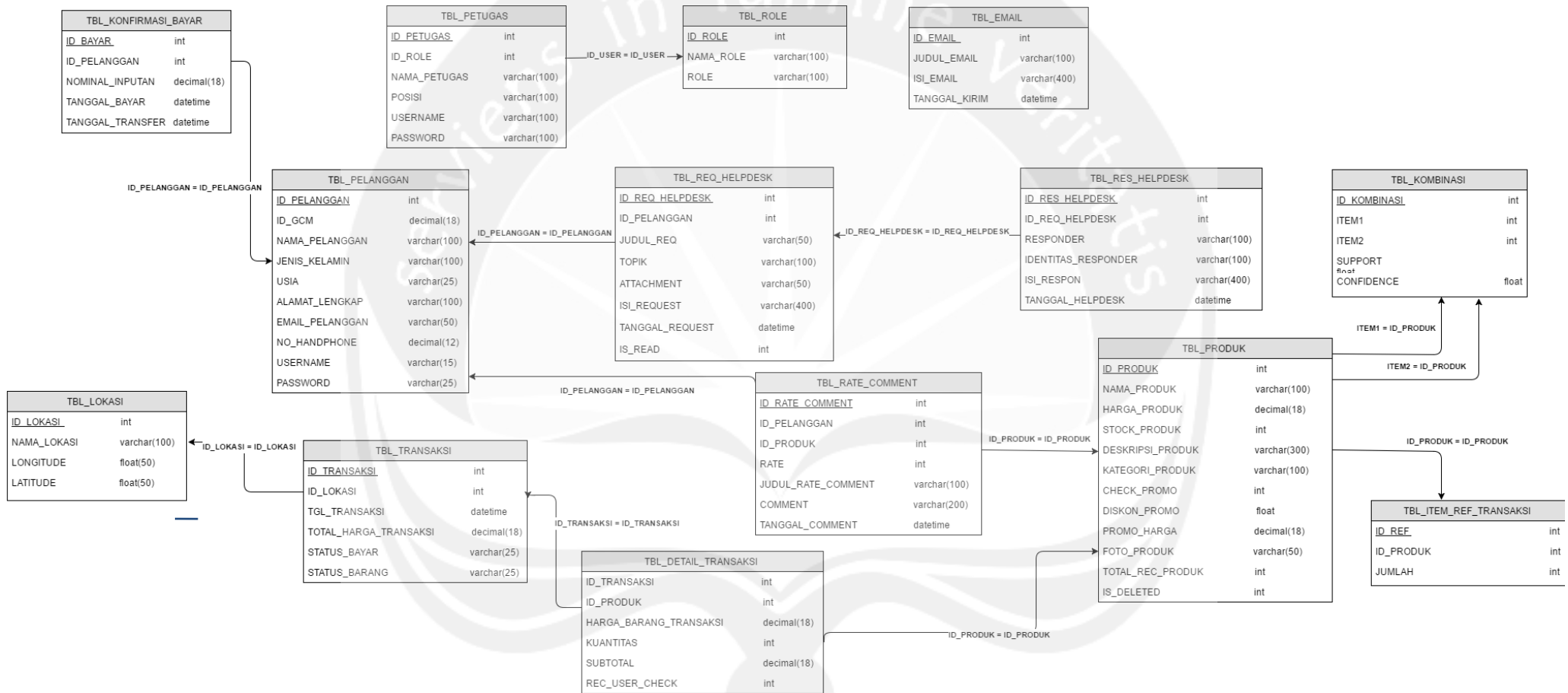
3.1.9 Deskripsi Entitas TBL_REKENING_PAMELLA

Nama	Tipe	Panjang	Keterangan
id_rekening_pamella	Integer	11	Id rekening milik pamella
nama_bank	Variabel Character	10	Nama bank dari rekening pamella
atas_nama	Variabel Character	100	Atas nama rekening pamella
no_rekening_pamella	Variabel Character	100	No rekening pamella

3.1.10 Deskripsi Entitas TBL_ROLE

Nama	Tipe	Panjang	Keterangan
id_role	Integer	11	Id dari role
nama_role	Variabel Character	30	Nama dari role

3.2 Physical Data Model



Gambar 3.1 Physical Data Model

4. Deskripsi Perancangan Antarmuka

4.1 Antarmuka Halaman Beranda



Gambar 4.1.1 Antarmuka beranda belum login



Gambar 4.1.2 Antarmuka beranda setelah login

Antarmuka digunakan sebagai halaman utama dari aplikasi ketika pertama kali diluncurkan. Terdapat dua tampilan yaitu gambar 4.1 yang menunjukkan gambar tampilan sebelum pengguna login sebagai pelanggan, dan gambar 4.2 yang menunjukkan gambar tampilan setelah pengguna login. Perbedaan ini terlihat dari tidak adanya tab menu produk laris pada tampilan login, sedangkan pada tampilan setelah login terdapat tab menu produk laris.

4.2 Antarmuka Halaman Login

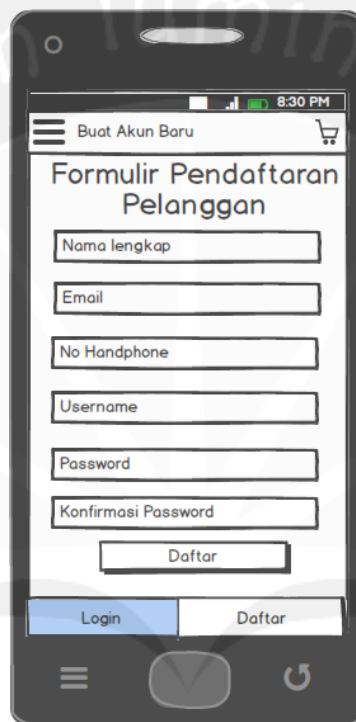


Gambar 4.2.1 Antarmuka halaman login

Antar muka ini digunakan untuk melakukan login pada aplikasi. Login ini terdiri dari input username, dan password. Jika username dan password cocok. Maka tampilan aplikasi akan kembali ke halaman

utama dengan menampilkan tampilan halamn utama setelah login. Jika username dan password tidak sama maka ditampilkan pesan bahwa username dan password tidak sama.

4.3 Antarmuka Halaman Daftar



Gambar 4.3.1 Antarmuka halaman daftar

Antar muka ini digunakan untuk mendaftarkan diri sebagai pelanggan / member pamella swalayan. Calon pelanggan mengisi formulir yang disediakan. Kemudian apabila sudah lengkap pelanggan menekan tombol daftar untuk mendaftarkan diri. Apabila ada formulir yang tidak di isi, maka aplikasi akan menampilkan pesan bahwa ada bagian formulir yang belum diisi.

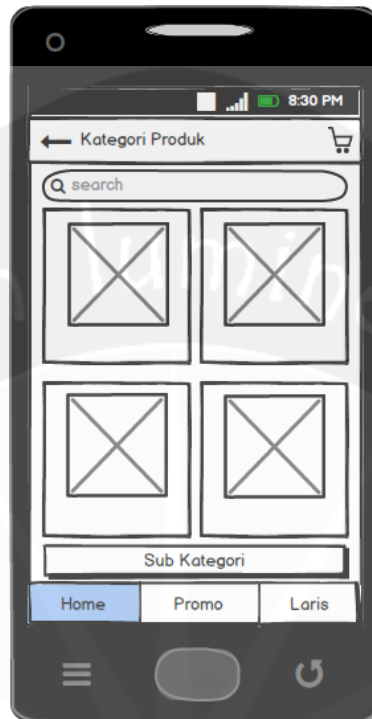
4.4 Antarmuka Menu



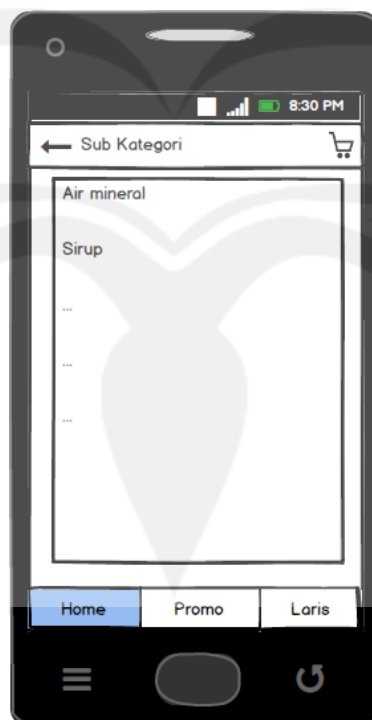
Gambar 4.4.1 Antarmuka menu

Antar muka ini digunakan pengguna sebagai navigasi selama menggunakan aplikasi. Tampilan menu ini terdiri dari beranda, kotak masuk, troli belanja, history transaksi, form pembayaran, tentang pamella mobile, dan keluar.

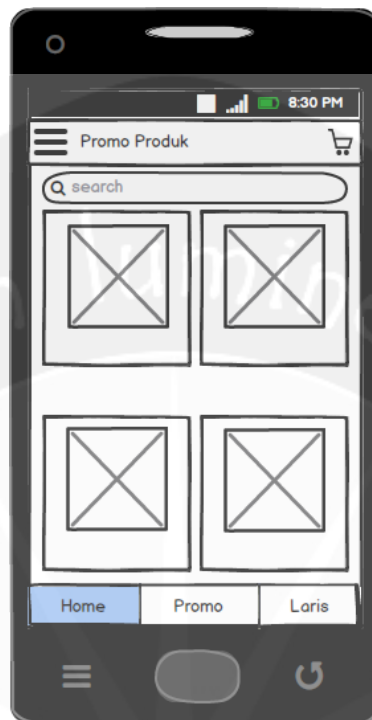
4.5 Antarmuka Tampil Produk



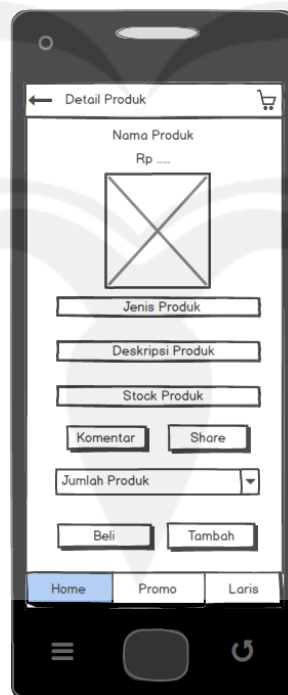
Gambar 4.6 Antarmuka tampil produk kategori



Gambar 4.7 Antarmuka tampil sub kategori



Gambar 4.8 Antarmuka tampil produk promo



Gambar 4.9 Antarmuka tampil detail produk

Antar muka ini digunakan untuk menampilkan produk yang terdapat di dalam database. Antar muka tampil produk terdiri dari beberapa bagian yaitu tampil produk kategori, tampil sub kategori, tampil produk promo, detail produk. Berikut adalah keterangan dari setiap antarmuka:

1. Tampil Produk Kategori

Antar muka ini menampilkan produk-produk yang dibagi menjadi beberapa bagian di dalam kategori.

2. Tampil Sub Kategori

Antar muka ini digunakan untuk menampilkan produk-produk yang di bagi menjadi sub kategori dari kategori.

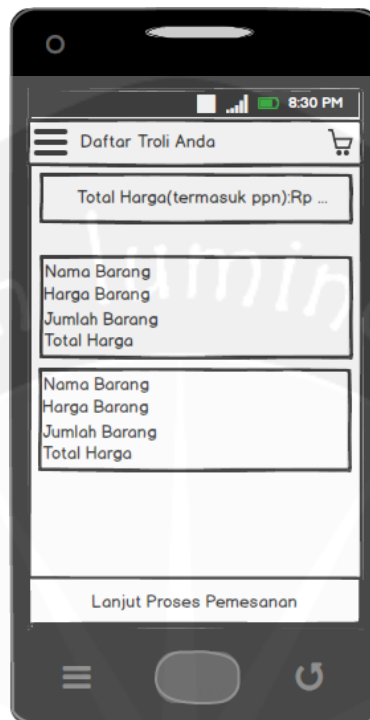
3. Tampil Produk Promo

Antar muka ini digunakan untuk menampilkan produk promo dengan menampilkan diskon dan potongan harga.

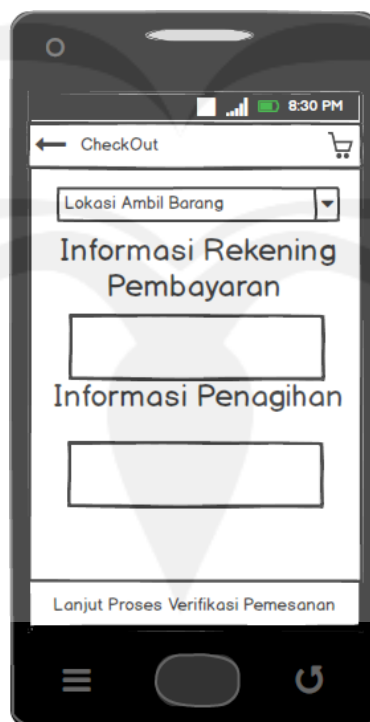
4. Tampil Detail Produk

Antar muka ini digunakan untuk menampilkan detail produk dari setiap produk yang dipilih. Selanjutnya dari detail produk, pengguna dapat memilih jumlah produk untuk dipesan.

4.6 Antarmuka Daftar Troli



Gambar 4.10 Antarmuka daftar login



Gambar 4.11 Antarmuka checkout



Gambar 4.12 Antarmuka verifikasi pemesanan

Antar muka ini menampilkan informasi mengenai produk yang telah dipilih oleh pelanggan sebelum dilakukan upload transaksi. Antar muka ini dibagi menjadi daftar troli, checkout, dan verifikasi pemesanan. Berikut penjelasan antar mukanya:

1. Daftar Troli

Antar muka ini menampilkan informasi keranjang belanja yang berisi list produk dengan total keseluruhan harga produk. Pengguna pada antar muka ini dapat menghapus produk didalam keranjang belanja, dan mengatur ulang jumlah produk.

2. Checkout

Antar muka ini digunakan untuk memilih lokasi pengambilan barang. Pada antar muka ini pengguna mendapatkan informasi mengenai rekening pembayaran dan pada siapakah penagihan ini ditunjukan.

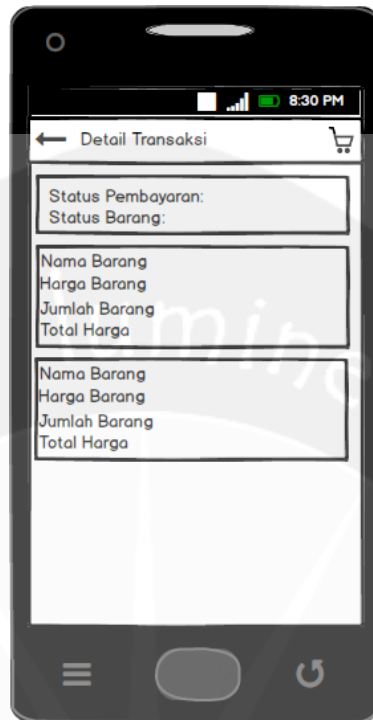
3. Verifikasi

Antar muka ini digunakan untuk verifikasi kembali data yang telah di isi oleh pengguna pada antar muka daftar troli, dan checkout. Antar muka ini bertujuan untuk memastikan kembali data yang di isi dan barang-barang yang akan di beli sudah benar.

4.7 Antarmuka Daftar History Transaksi



Gambar 4.13 Antarmuka daftar history pemesanan



Gambar 4.14 Antarmuka daftar detail transaksi

Antar muka ini digunakan untuk menampilkan informasi dari transaksi yang dilakukan oleh pengguna. Informasi ini terbagi menjadi 2 bagian yaitu. Antar muka daftar transaksi, dan antar muka daftar detail transaksi.

4.8 Antarmuka Formulir Pembayaran

Formulir Pembayaran

Nama Pelanggan

ID Transaksi

Informasi Rekening Pembayaran

Pilih Rekening

Kirim

Login Daftar

Gambar 4.15 Antarmuka formulir pembayaran

Antar muka ini digunakan untuk mengirimkan informasi pembayaran. Pelanggan yang sudah melakukan pembayaran, wajib mengirimkan informasi pembayaran yang telah dilakukan. Pelanggan memilih id transaksi yang belum dibayar dan milik tujuan dari rekening. Selanjutnya mengirim informasi tersebut dengan menekan kirim.

4.9 Antarmuka Web Login



A Web Page

http://localhost/pamellaswalayan

Selamat Datang, silahkan login

Username

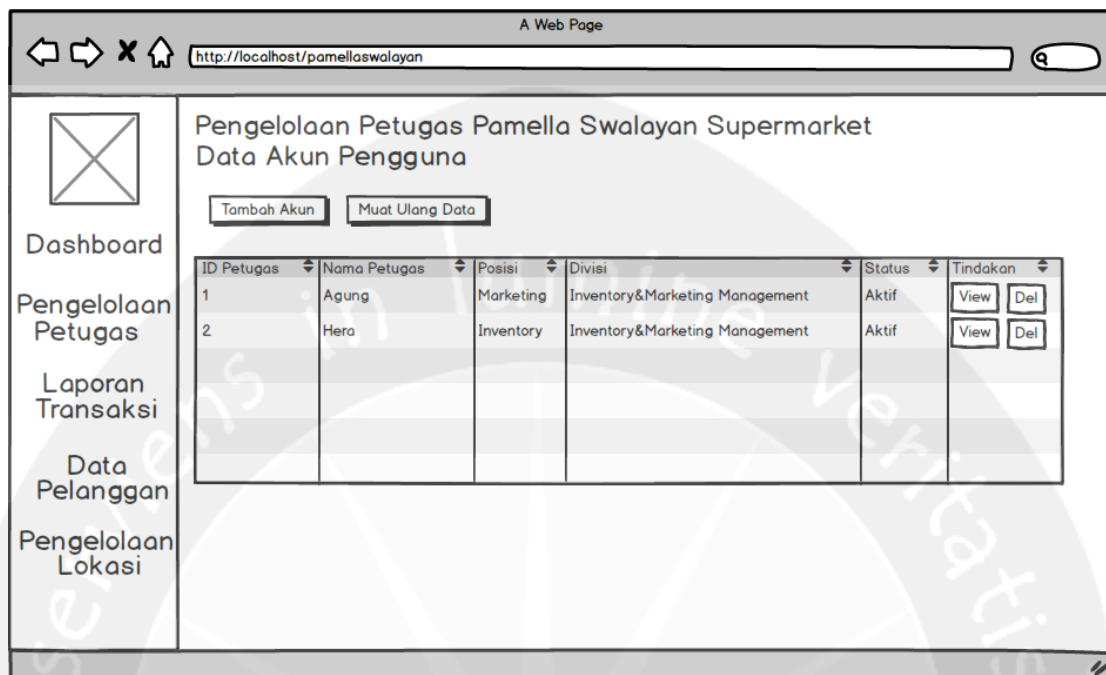
Password

Verifikasi Akses Masuk

Masuk


Antarmuka ini digunakan sebagai halaman pembuka ketika user mengakses web pengelolaan. User memasukan username dan password kemudian sistem akan melakukan validasi pengecekan username dan password. Apabila salah, akan ditampilkan pemberitahuan bahwa username dan password salah. Jika benar, maka user akan dibawa ke halaman pengelolaan sesuai dengan role user.

4.10 Antarmuka Web Pengelolaan Petugas



Antarmuka ini digunakan untuk melakukan pengelolaan web untuk petugas. Admin dapat menambah petugas baru, mengedit informasi dari petugas, maupun menghapus petugas.

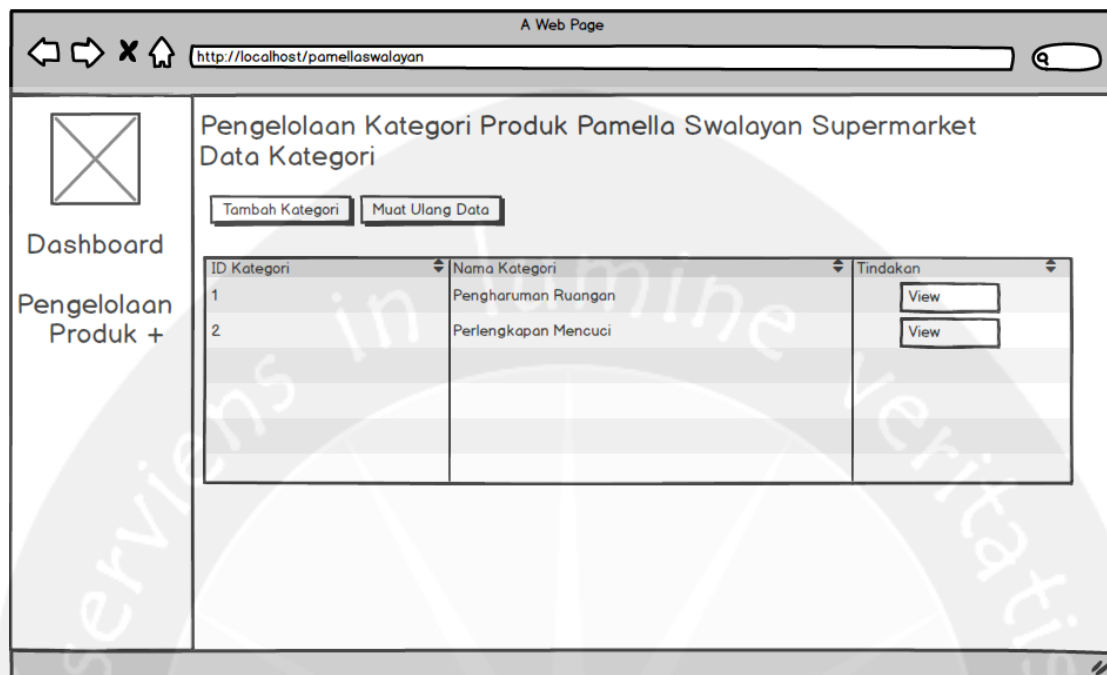
4.11 Antarmuka Web Verifikasi Akun Petugas



The screenshot shows a web browser window with the address bar displaying "http://localhost/pamellaswalayan". The page content includes a large rectangular box with a diagonal 'X' inside, indicating a missing image or a placeholder. Below this, there is a form titled "Silahkan Verifikasi Akun Anda". The form contains two input fields: "Id" and "Nama lengkap". Below these fields are two buttons: "Cek Akun Petugas" and "Kembali ke Login".

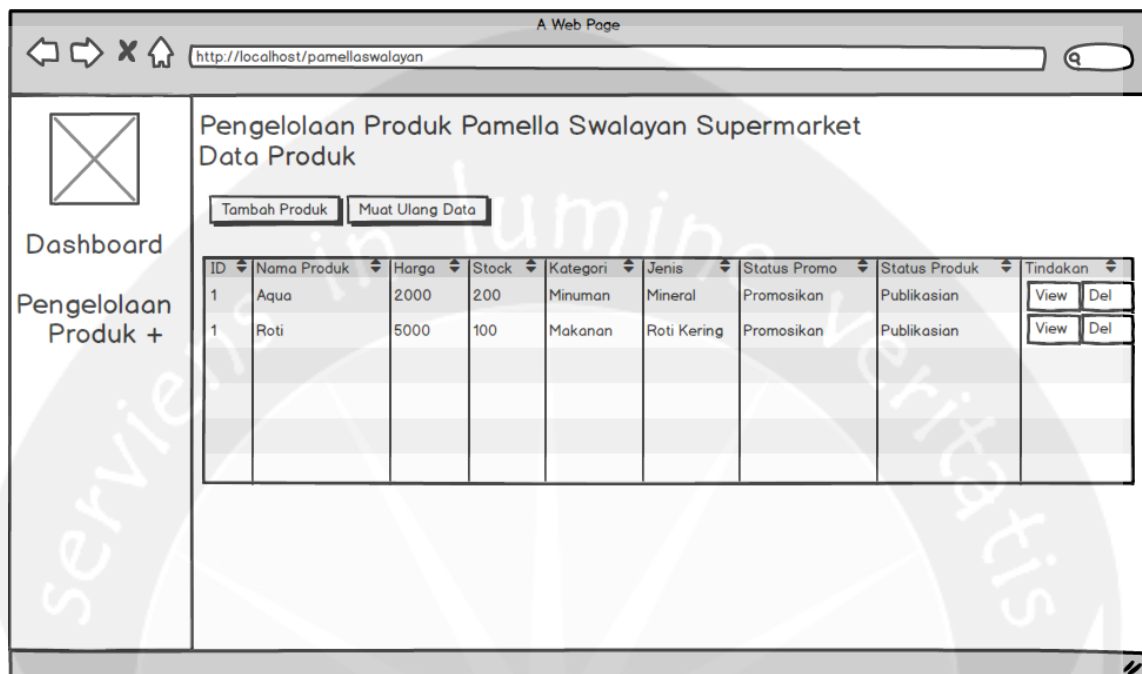
Antarmuka ini digunakan untuk melakukan pengecekan validasi pada akun petugas yang telah dibuat oleh admin. Sistem akan melakukan pengecekan terhadap id yang telah dibuat oleh admin dan nama lengkap petugas. Apabila tidak cocok, sistem akan menampilkan informasi bahwa kedua inputan tidak cocok. Apabila cocok maka, petugas diminta untuk menginputkan username dan password.

4.12 Antarmuka Web Pengelolaan Kategori



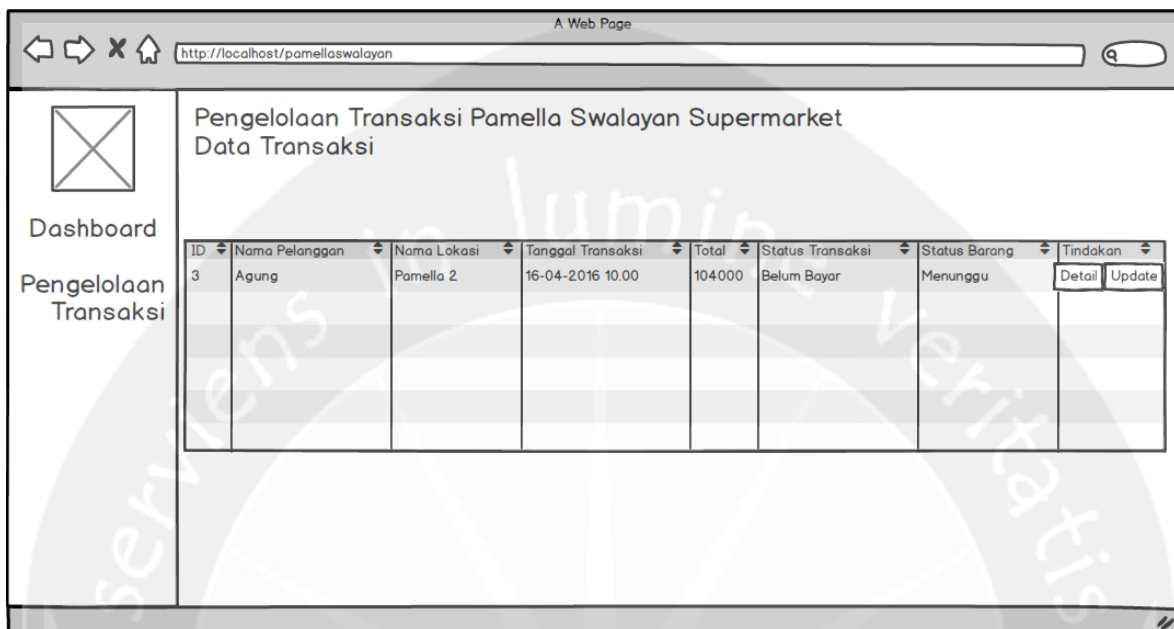
Antarmuka ini digunakan untuk melakukan pengelolaan terhadap kategori produk. Petugas dapat menambah kategori dan melakukan perubahan terhadap nama kategori.

4.13 Antarmuka Web Pengelolaan Produk



Antarmuka ini digunakan untuk melakukan pengelolaan terhadap produk pamella swalayan. Petugas dapat melakukan penambahan produk dan melakukan perubahan terhadap informasi produk. Petugas juga dapat menghapus produk sehingga tidak ditampilkan dan dilihat oleh pengguna aplikasi mobile.

4.14 Antarmuka Web Pengelolaan Transaksi



Antarmuka ini digunakan untuk melakukan pengelolaan terhadap transaksi yang telah dilakukan oleh pelanggan. Petugas dapat melihat detail dari transaksi dan melakukan perubahan status pembayaran dan status barang.